An Interop Publication

CONNEXIONS

The Interoperability Report

September 1993 | *Special Issue: INTEROP Companion* | Volume 7, No. 9

ConneXions — The Interoperability Report tracks current and emerging standards and technologies within the computer and communications industry.

In this issue:

NSFNET Celebrates 5 years...2
Standards the IETF way.......18
Remote Printing.................... 27
CIDR & the evolution of IP... 30
A User's View of IPng............36
The Internet for Business..... 41
Internet Security................... 46
Internet Services................... 50
ATM For Your internet.........56
GateD.....................................61
Introduction to Routing.........66
Gigabit Network Interfaces...74
Windows Sockets...................80
Wireless Networking.............87
CALS.....................................90
Announcements.....................95

ConneXions is published monthly by Interop Company, 480 San Antonio Road, Mountain View, CA 94040–1219, USA. +1 415-941-3399. Fax: +1 415-949-1779. Toll-free: 1-800-INTEROP. E-mail: `connexions@interop.com`.

ISSN 0894–5926

From the Editor

Welcome to INTEROP in San Francisco and to this Special Issue of *ConneXions—The Interoperability Report.* This edition contains articles directly and indirectly related to the conference, with pointers to sessions, tutorials and BOFs at the end of each article.

A major focus of INTEROP and of this issue is the *Internet,* the rapidly expanding, world-wide system with nearly 2 million host computers. Our first article is on the NSFNET backbone, an important Internet component. The NSFNET celebrates its fifth anniversary this year.

INTEROP 93
23–27 August 1993
Moscone Convention Center
San Francisco, CA
AUGUST

We then look at the Internet from a number of perspectives. First, Dave Crocker describes the rather unique (and very successful) standards mechanism. This is followed by an article on a recent Internet experiment, *Remote Printing.* Next we have two articles on the future of the *Internet Protocol* (IP). The first article is by a team of protocol developers, the second presents the point of view of an end user.

The Internet is already much more than an academic and research network. Businesses are starting to rely on Internet service much as they rely on telephone service. John Curran examines some of the issues facing a commercial user of the Internet. Of great concern to any business is *security,* so we naturally follow this article with one by Jeff Schiller entitled "Issues in Internet Security." Any user of the Internet will also want to know what services it can offer, and Peter Deutsch has the answer in his Internet "tools" overview. Next, Mark Laubach looks at ATM and how/when it might affect Internet users.

Routing continues to be a "hot topic" in the Internet community and elsewhere. Martyne Hallgren and Jeff Honig present *GateD,* a widely used routing system for the Internet. This is followed by an introduction to routing, by Dave Piscitello and Lyman Chapin.

Craig Partridge outlines the design and implementation of gigabit network interfaces. Martin Hall describes *Windows Sockets* which grew out of an INTEROP BOF session a couple of years ago. Rifaat Dayem gives us a roadmap to wireless networking and mobile computing. All are topics you will hear much more about at INTEROP, and that you can continue to read about in *ConneXions,* so sign up!

Finally, a look at CALS, *Computer-aided Acquisition and Logistic Support,* a DoD and industry strategy to transition to automated, integrated product development design, manufacturing, and support processes. The article is by Carolyn Wimple.

The NSFNET T1/T3 Network

Backbone of Emerging Information Infrastructure Celebrates 5 Years of Extraordinary Growth (1988–1993)

by Paul D. Bosco, Massachusetts Institute of Technology and Hans-Werner Braun, San Diego Supercomputer Center

Introduction

The National Science Foundation's high speed (T1/T3) interstate network, NSFNET, celebrates five years of operation this summer. From 1988 to 1993, NSFNET has grown explosively with profound impact on US research and education. NSFNET's success established US leadership in national networking and provided the basis for the *National Research and Education Network* (NREN) Program. NREN will vastly broaden the US *National Information Infrastructure* (NII). The success of NSFNET has also spawned new firms, and encouraged established firms to offer new network services. NSFNET was an early user of the technologies being demonstrated and discussed at INTEROP.

Growth

NSFNET's growth has been extraordinary. In the Summer of 1988, a new highway for US science and education, the high speed NSFNET backbone, was opened for use. Since 1988, traffic has increased several orders of magnitude, from millions of packets (bundles of data carried across the network) to billions per day. The number of colleges, universities and research laboratories using NSFNET has increased from less than 200 in 1988 to more than 10,000 in 1993. Today, every major US research university, and the vast majority of four year colleges, use the NSFNET national backbone. Health care, K–12 and community college use has also begun growing quickly.

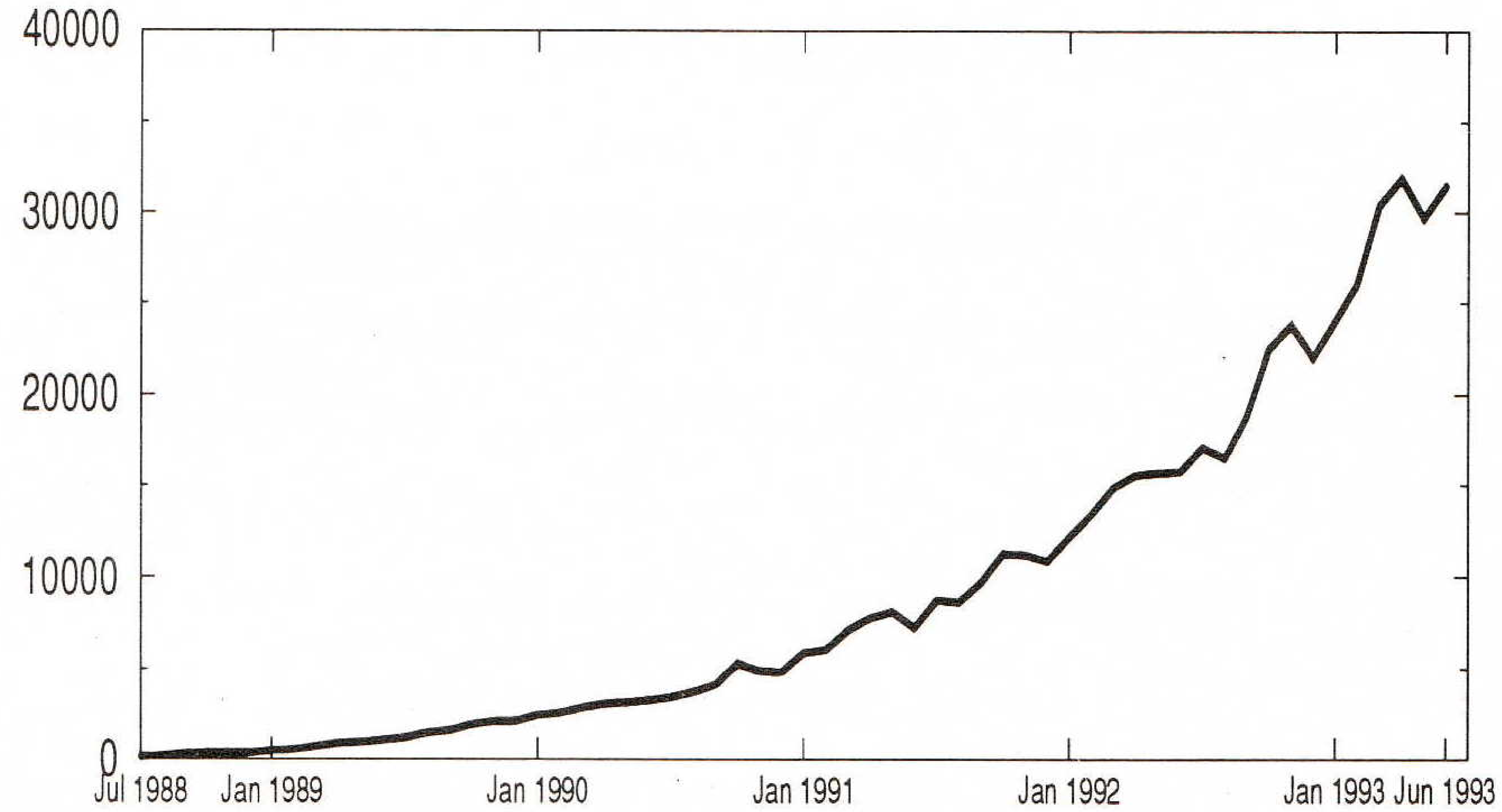


Figure 1: Megapackets per month on the NSFNET

Government and business

The NSFNET High Speed Backbone Project also demonstrated that government, at a state and federal level, can work with US businesses to share investment costs on major science and technology projects. For example, the State of Michigan contributed $1 million per year to complement National Science Foundation support. Major corporations, such as original NSFNET partners IBM and MCI, have also complemented NSF funding, providing contributions that significantly exceeded the NSF contribution. As NSFNET's inter-state use has exploded, new intrastate networks have emerged. State governments and their corporate partners have announced major network initiatives in Iowa, California, Georgia, North Carolina, etc. to enhance research, education and health—and provide local access to the interstate data highway.

Benefits

Small companies and the US high technology industry have also benefited from the NSFNET project. T3Plus, a California based company founded in 1989, utilized NSFNET developed technology otherwise unaffordable for a startup firm. T3Plus has grown quickly, helping establish US leadership in high speed networking products and adding 35% more employees each year. Larger high tech firms such as Intel have also benefited from NSFNET, collaborating with the NSFNET R&D team to accelerate new processor technologies for use in NSFNET's advanced networking applications. Building on ARPA sponsored Internet research, and NSFNET's technical development and advances, an entire billion dollar US Internetworking industry has emerged (the Internet is the mesh of interconnected networks allowing connectivity, remote access, electronic mail and information transfer). Some of America's fastest growing corporations (such as Wellfleet, Cisco and others) have been stimulated by the NSFNET investment in information infrastructure.

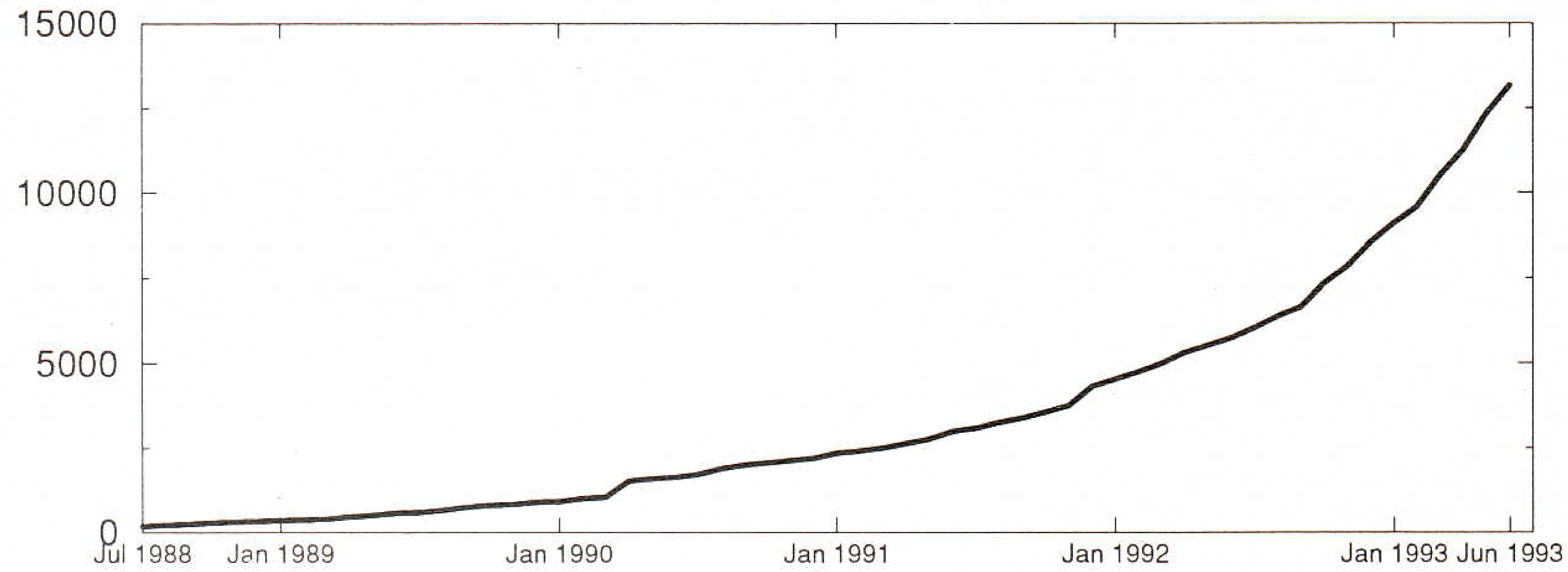


Figure 2: Network numbers known to the NSFNET

Applications

NSFNET helped changed the way we use networks, from providing local connectivity across an office building to nationwide information and knowledge sharing. In 1988, NSFNET was envisioned as a mechanism to provide nationwide access to national resources, most specifically federally funded supercomputer centers. While the centers still contribute major traffic flows, the NSFNET backbone's scope has expanded to interconnect the broader research and education community.

Today, NSFNET interconnects an even more important resource—*people*. The majority of users are engaged in collaboration and information access, instead of logging into remote supercomputers,. New network applications are enhancing benefits, such as tools providing near instantaneous access to networked information and experts—including electronic directories of networked resources and information nationwide. The network is used to instantaneously disseminate technical advances nationwide, and allows for interactive dialog among scientists and educators. NSFNET has begun carrying video and audio, from technical meetings to Presidential addresses (such as the Silicon Graphics address). New tools such as electronic white boards allow a speaker's notes to be shared in real time with viewers over a broad area. As an experiment in national networking, NSFNET has changed research and education forever, and continues to increase in scope and impact.

This year, the NSFNET high speed backbone will be enhanced again. New directory services, new information tools and higher speed fiber optic links will become available. The network's capacity will continue to increase, from megabits per second to gigabits per second. In addition, an open network architecture will encourage all US telecommunications providers (large and small) to offer connectivity services.

continued on next page

The NSFNET T1/T3 Network *(continued)*

A bright future

This should result in increased access to the nation. A rich array of new multimedia services, in exciting fields such as health, K–12 and science and technology will emerge. Corporations continue to build on NSFNET's leadership, with companies such as Sprint now offering Internet services worldwide to commercial users. As the T1/T3 NSFNET Backbone Project celebrates 5 years of operation, the future of US networking is bright.

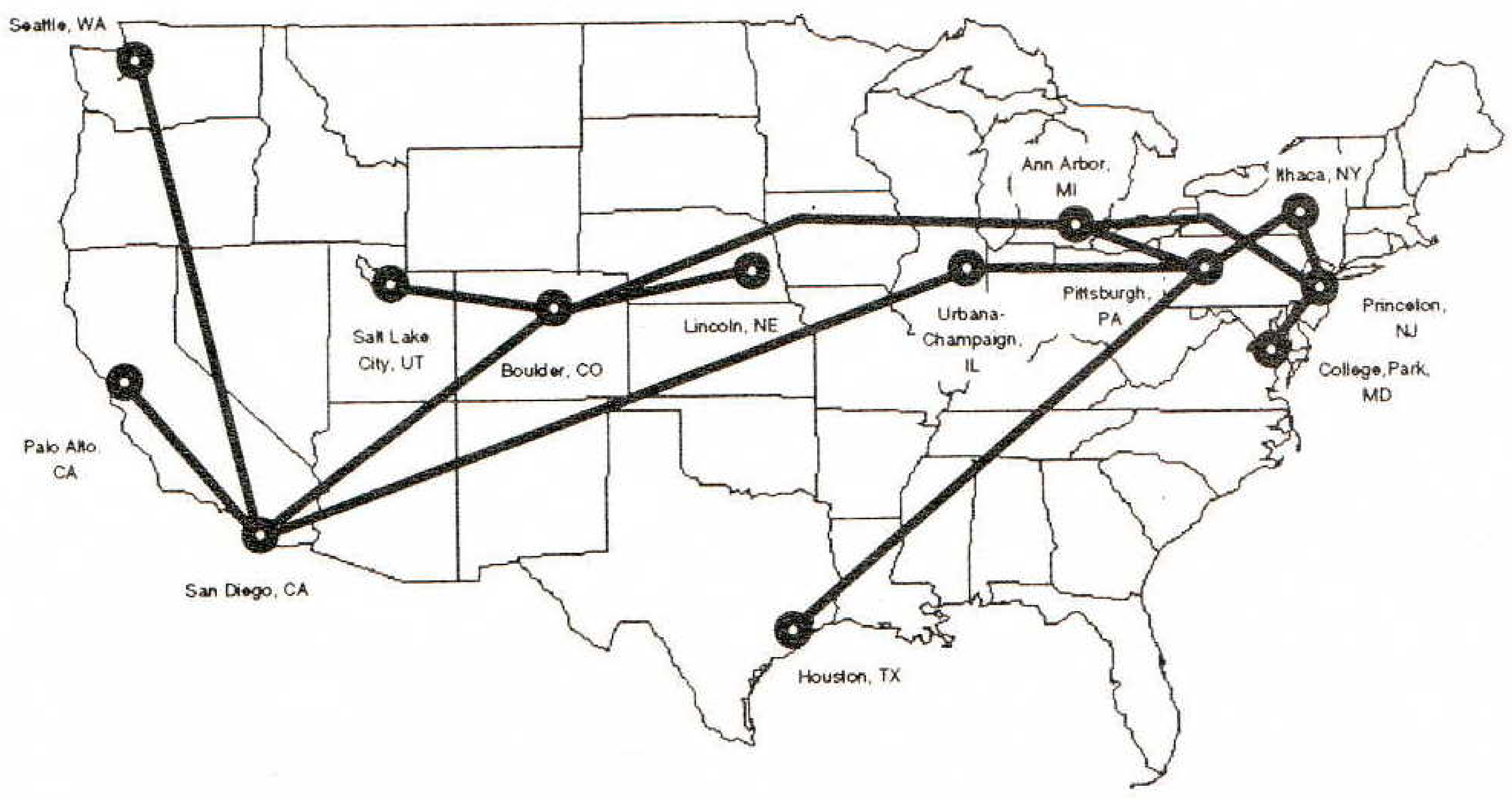


Figure 3: Initial T1 NSFNET Backbone, July 1988

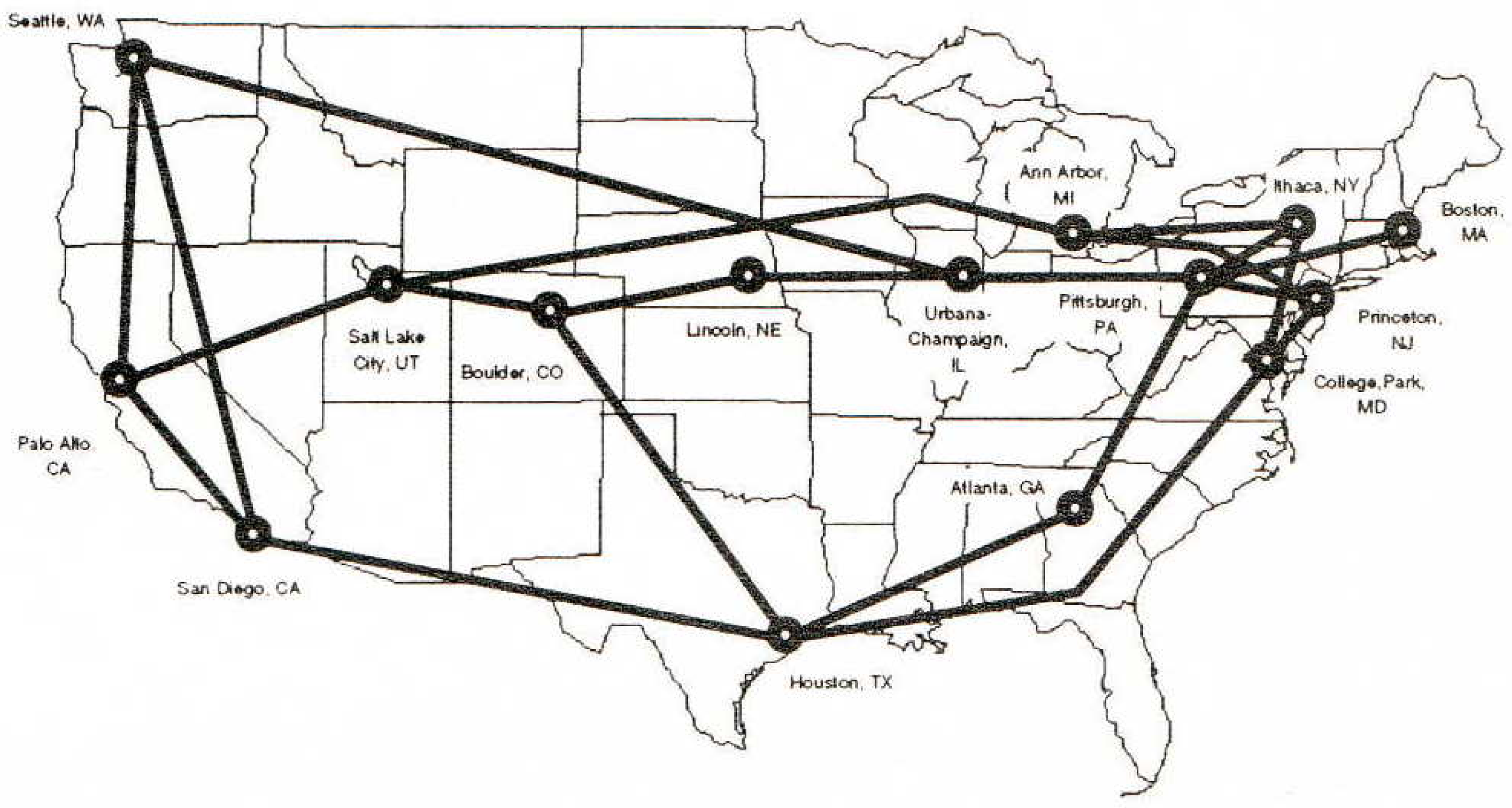


Figure 4: Reconfigured T1 NSFNET Backbone, July 1989

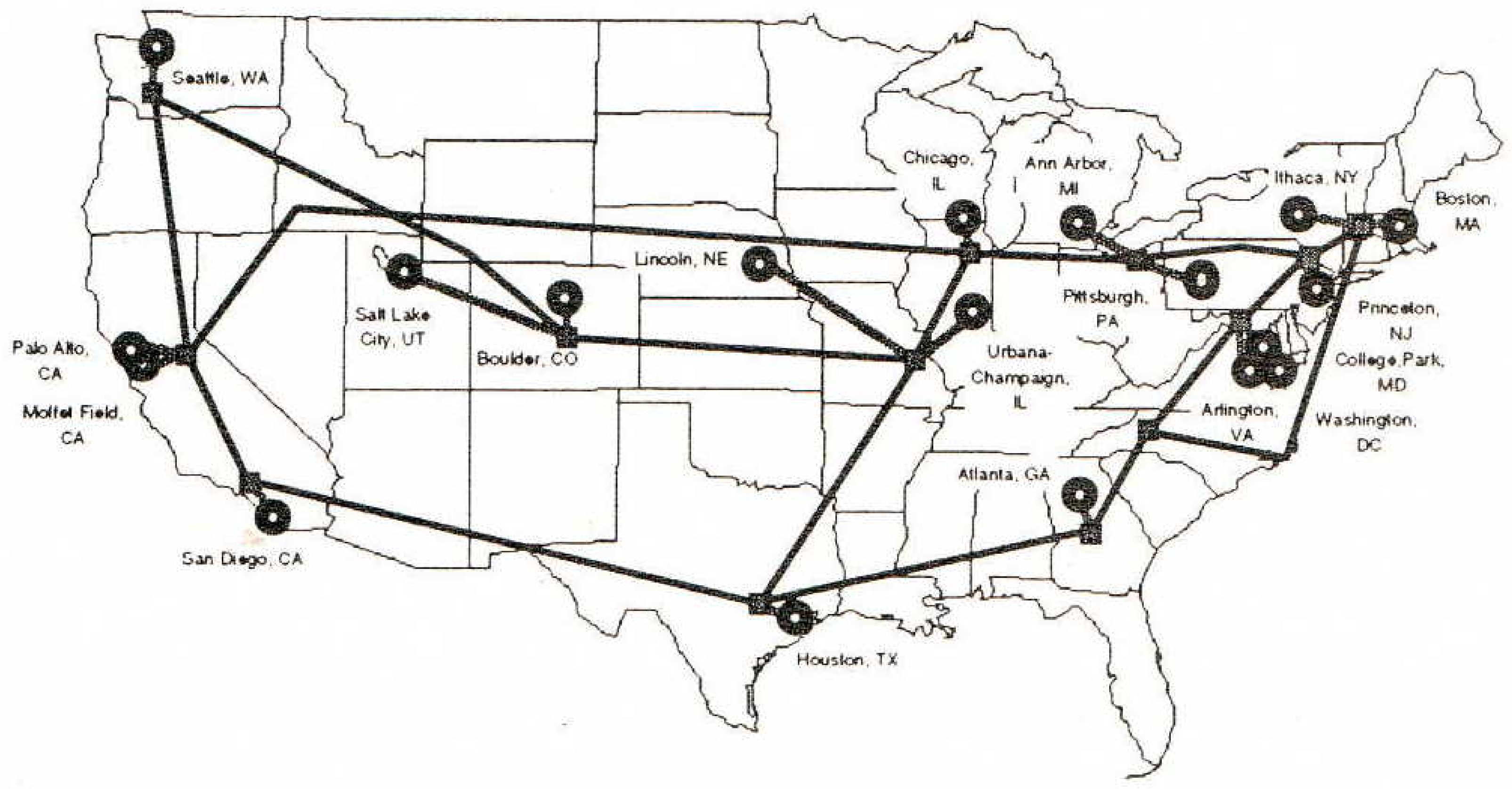


Figure 5: T3 NSFNET Network Service over ANSnet start in January 1991

HANS-WERNER BRAUN joined the San Diego Supercomputer Center (SDSC) as a Principal Scientist in January 1991, focusing on NSF funded efforts for NREN engineering and network performance related research. His prior work included being a Principal Investigator on the NSFNET backbone project while working at Merit, Inc., in Michigan. After graduating in 1978 in West Germany, holding a Diploma in Engineering he worked for several years at the Regional Computing Center of the University of Cologne in West Germany on network engineering. He participates in several national and international networking related committees. His e-mail address is: `hwb@sdsc.edu`

PAUL D. BOSCO holds a B.S. in Electrical Engineering from Lehigh University, M.S. in Computer Engineering from Yale University and MBA/M.S. from Rensselaer Polytechnic Institute. He is currently pursuing a PhD at the Massachusetts Institute of Technology and participating in research on networked multimedia technology and services for science and technology, health care, and K–12. From 1987 to 1992 he organized and led IBM's systems and technology group on the NSFNET effort. He previously held programming, operations and management positions for IBM. He can be reached at: `bosco@mit.edu`

NSFNET Interviews

by Richard Solomon, Massachusetts Institute of Technology

ConneXions captured the comments of key NSFNET visionaries and technical leaders. Each interviewee was asked to comment on NSFNET and the future of networking. Their comments follow.

- Vice President Albert Gore (`vice.president@whitehouse.gov`)

Mr. Vice President, you have played a key role in national networking. Can you comment on your efforts supporting NSFNET and its impact on the emerging National Information Infrastructure?

Wide impact

"NSFNET shows both how government and business can work together to promote advances in computing and networking and how much impact a small amount of government funding in the right place can have. The original idea was for the NSF backbone to connect scientists who needed access to supercomputers. Since it made so much sense for scholars and campus network administrators to link their computing and information resources, the NSFNET has grown into the key component of the Internet, which today includes more than 7000 domestic networks throughout the US. Through the creation of internetworking standards and a modest, although essential investment in the backbone, we have been able to speed up the research and discovery process and create dozens of new network services companies. We think the Internet can have an impact far beyond the research and educational communities and help lay the foundation for the NII."

What is your long-term vision for the National Information Infrastructure? How do you envision it in five years from now?

"Past policies have made the two key electronic devices for mass communications and entertainment—the television and the telephone—virtually ubiquitous. Improvement of the NII coupled with intelligent government policies, mean that future generations of information appliances and services they can provide will also be widely available. Citizens will have greater control over the information they choose to receive and the form in which they wish to receive it. In facilitating private sector innovation and investment in these areas we will defend the principles that are at the very core of the idea of American commitments to freedom of speech, privacy and citizen participation and control over public affairs."

Has the extraordinary growth in the use of the NSFNET national backbone surprised you?

"Well, I don't think that anyone would have predicted the kind of growth we've seen. I think it shows how fast a good idea can travel when the old ideas aren't allowed to get in the way."

Do you have any comments for the NSFNET team as they celebrate the fifth year of the T1/T3 NSFNET?

Rewards

"Like the explorers of the New World 500 years ago, you are making new connections between different worlds, the effects of which none of us can know at present. When you read five years from now about new breakthroughs in modeling climate change, in medical imaging and remote diagnosis, or in the synthesis of complex chemical substances, you should understand that these achievements would not have been possible—or would have taken years longer—had it not been for your work. This is truly the most satisfying reward that anyone can ever have."

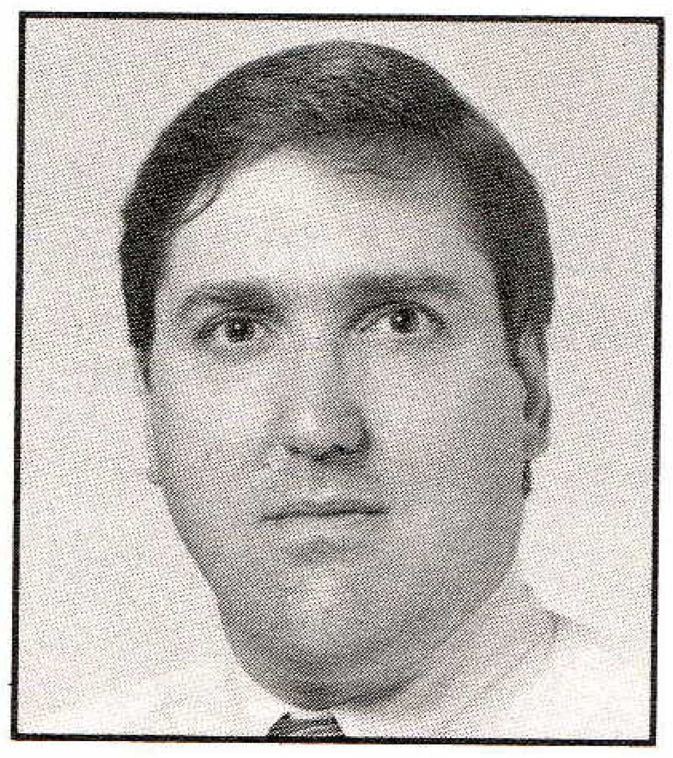
continued on next page

NSFNET Interviews *(continued)*

Vice President, five years ago the notions of a national data superhighway were not widely held. Today articles appear about this and the NII almost everywhere. How does this make you feel?

More to come

"Well, again I think it shows the power of a good idea more than the sponsorship and support of any one individual. Many, many people have been crucial in popularizing computer networking and in raising the national consciousness about information infrastructure. But as far as it may seem that we have come in such a short time, I believe that our most stunning achievements in using technology as a tool of empowerment still lie ahead of us."



- Paul D. Bosco (`bosco@mit.edu`)

What were your roles on the NSFNET T1/T3 project?

"I was the first IBM'er to join the T1/T3 NSFNET effort full time. I flew to Ann Arbor, MI in 1987 to participate in the NSFNET proposal development and more or less 'never' returned. I became IBM's lead NSFNET systems and technology architect, and organized and managed the team that developed several generations of router technologies. Every electronic letter, file transfer, etc. on NSFNET passed through hardware and/or software developed by that group."

What were the biggest challenges and surprises?

"Building a nationwide T1 TCP/IP network with 1987 technology was the first challenge. In 1987, there were no router products as we know them today, TCP/IP was not widely accepted outside the scientific and technical community; SNMP was still emerging; OSI/CLNP was the 'coming' protocol; and T1 data networks were somewhat new. Notably, the first INTEROP conference with an exhibition was not held until 1988, and only several thousand visitors attended (versus 55,000 at INTEROP 92, Fall)."

Always a race

"Undoubtedly, the greatest challenge has been staying ahead of NSFNET's extraordinary growth in traffic and networks. The growth has also been the biggest surprise. If we had declared in 1987 that NSFNET's routers be capable of handling 10,000+ networks they would have had us committed. Yet today, 100,000 route scenarios must be seriously considered. Also, with 10%, 20% or more traffic growth per month, there was absolutely no room for error. New capacity had to be bought on line and technology transitions managed carefully without interrupting service. Over my four years on NSFNET, there wasn't a moment to catch our breath. It was always a race to stay ahead of traffic and addresses."

"In retrospect, we were very fortunate the core NSFNET team (Merit and corporate partners), other agencies, regionals, and campus network folks remained focused and worked together so well. The cooperation was unimaginable, an outstanding example of a federally sponsored program that captured and focused incredible human energy. Each of us on the NSFNET team has been fortunate to have an inside view as the Internet exploded and an entire industry was born. It was an extraordinary experience with constant challenges and surprises."

What have we learned?

"We learned to ignore those who said it would never work, others who assured us TCP/IP would not scale, and race forward."

"We learned routing technologies could deliver nationwide connectivity and that users would become critically dependent on those connectivity services. We learned maintaining high network availability required more sophisticated management tools than we first imagined and that intense effort is required to develop management tools that provide genuine leverage. We've certainly learned that a national infrastructure does not require centrally planned decision making, and may progress more quickly in a distributed fashion."

What are some of the technical lessons?

The importance of tools

"Design for reliability, availability, manageability, and integrate statistical collection tools from day one. Our real trick was continuously upgrading technology and managing those technology transitions through periods of intense network growth. Without the statistical and management tools used on NSFNET, the network would have surely collapsed. As the network infrastructure transitions again to higher speed ATM technologies, we need to continue focusing on reliability, availability and manageability."

What are your fondest memories or incidents?

"In the summer of 1988, the T1 NSFNET came on line smoothly and the performance improvements were dramatic. Users expressed astonishment at the ability to send electronic mail reliably within seconds, nationwide. Coast to coast Telnet sessions became usable; large files began being exchanged regularly; SNMP began to come together; and ESF began providing excellent diagnostic capability on some links. 1988 was an electric period. It made the 7 day work weeks and 16 hour days worth it. Of course, we were too naive to understand our users' enthusiasm was a key indicator of the future growth (and sleepless nights) to come."

The perils of hunting

"One memorable incident was our scramble in 1989 (I believe) to recover from hunters shooting out the fiber carrying NSFNET over a swamp. The MCI team had prepared us for circuit losses by repeatedly warning us that fiber has a peculiar physical property which attracts backhoes. But the swamp shootings were unexpected."

What are the next challenges?

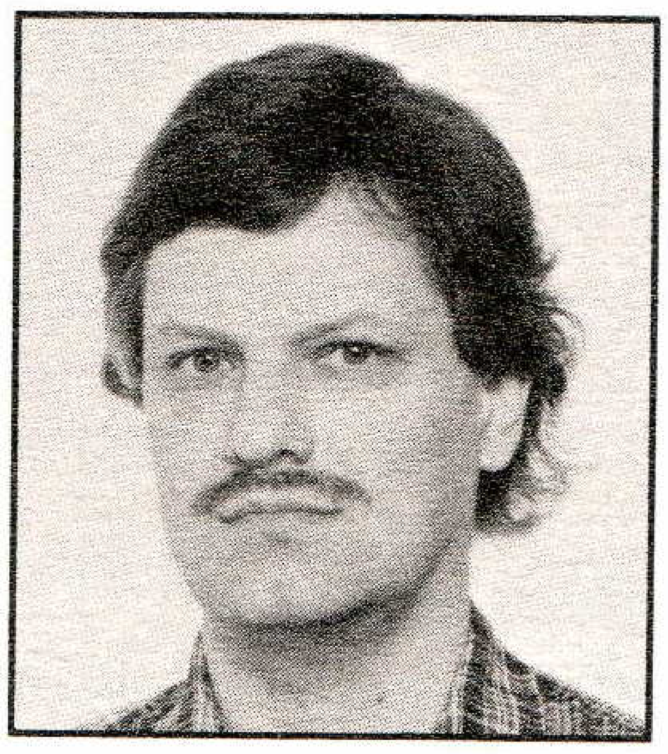
"My focus will be expanding our vision of the Internet, to include new players such as cable firms; achieve video interoperability between Internet, telco and cable; and deliver new Internet services such as networked multimedia health care, Telemedicine, and information services. It seems the next generation of network, video, server and set top box technologies can deliver new services that provide profound business and social benefits. So I will personally be working with organizations, and involved in research, in those areas."

ATM

"The next 12 months are a critical time for the National Information Infrastructure. NSFNET will be replaced, and a large number of commercial Internet providers will emerge. ATM technologies will be more widely deployed, and new multimedia services (connectivity, distribution and application services) will emerge. The TCP/IP to ATM migration or coexistence also needs to be settled. As our network infrastructure becomes more diverse, preserving the shared objectives and forward momentum NSFNET helped establish will be the greatest challenge."

Paul Bosco was awarded a doctoral fellowship in 1992 and is currently pursuing a PhD at the Massachusetts Institute of Technology.

continued on next page



NSFNET Interviews *(continued)*

- Hans Werner Braun (hwb@sdsc.edu)

What was your role on the NSFNET T1/T3 project?

"I was Principal Investigator and played an overall coordination and management role for the project. Much of my time was spent coordinating the project with the NSFNET backbone partnership, including with the National Science Foundation. I was furthermore manager of Merit's Internet Engineering group. Before the time of the T1/T3 NSFNET backbone I also played a critical role in making the previous 56kbps NSFNET backbone work and to keep it running."

What were the biggest challenges?

Building trust

"The major challenges were both technical and non-technical. Obviously the technology challenges were critical. However, the greatest challenge was harnessing the Internet community to build a large operational infrastructure, trust its qualities, and constantly advance its performance and ubiquity. Many constituencies were quite critical of the NSFNET award and the probability of the project's success. Some expressed very severe doubts a nationwide data network, NSFNET, would ever function as proposed. We had to build up trust, but that came as we delivered a sound infrastructure. The communication, management and administration of the project were a quite serious issue."

What were the biggest surprises?

"For me the biggest surprise was the rather pleasant working relationship with big-US-industry. They were very interested in new concepts and new ideas and incredibly supportive of the project. It was much more of a pleasure working with IBM and MCI than I had ever expected."

What have we learned?

Government's role

"We have certainly learned the United States Government is still able to push the high technology envelope, helping to ensure US competitiveness. Often it is argued government science and technology policies work better in other countries, but data communication is one of few high tech areas the US leads *because* the government plays an active role and has stimulating its evolution. National and international leadership does not happen by accident."

What technical advances did NSFNET create or stimulate?

"The NSFNET was key in the creation of a multibillion dollar data communications industry and triggering further development in a myriad of areas, from routing, switching and transmission technologies to advanced networking environments and applications which are changing the way people interact and communicate."

What interesting NSFNET memories can you share?

"There are really so many memories, events and experiences that it is very hard to articulate a good set with limited time or space. It was, however, particularly gratifying having been involved in the national backbone's evolution from the original 56kbps NSFNET backbone to the first cross country T3 operational demonstration by the end of 1990. It has been interesting watching the move towards more commercialized and industrial-strength network environments."

What are the next challenges?

"The principal issue with the current infrastructure is the need to develop and maintain an industrial-strength, commercialized environment, that can be largely invisible to network users. The underlying communication fabric needs eventually be viewed as 'a given,' with the new focus being in the areas of information and services that utilize the infrastructure. Many of the currently used applications on the network remain primitive at best, and there is a tremendous opportunity for improvement in usability and benefits."

Where do you see NSFNET, NREN and the National Information Infrastructure in 3–5 years?

Access for everyone

"Besides the need for advanced applications, someone has to ensure ubiquitous communication is possible. I think we have a reasonably good handle on high performance networking, getting to high speeds, and such. What is still lacking are the capillary connections, or local access links, that will eventually ensure everyone an ability to communicate with everyone else throughout the United States. This takes real investment, something that is beyond the scope of the federal government. It will require broad collaboration from local governments, as well as significant investments by United States industry. As such, I am glad to see, for example, major telephone companies, principal providers of ubiquitous infrastructure, getting more and more interested in data communications. Some of this is gated by some mix of technical and administrative issues, like network accounting, for example, to ensure a graceful migration towards more comprehensive and commercialized environments. Taking these things together, I would expect a significantly increased participation by industry and local government to enhance the current interconnected infrastructure, as well as significant improvements in advanced applications, making people–people, as well as people–machine interactions very sophisticated."

Hans-Werner Braun is currently a Principal Scientist at the San Diego Supercomputer Center (SDSC) focusing on NSF funded efforts for NREN engineering and network performance related research.

- Eric Aupperle (`Eric.M.Aupperle@um.cc.umich.edu`)

What was your role in the NSFNET T1/T3 project?

"Merit, with IBM and MCI as our joint study partners, won the 1987 NSFNET Cooperative Agreement. I've been involved with the NSFNET activity from the proposal writing stage to the present time. I am the Project Director for NSFNET."

What were the biggest challenges and surprises?

"There have been many challenges. Among them was to assemble the team among Merit, IBM and MCI to implement the T1 network on the proposed schedule, i.e., within seven months of the award. We did that very successfully. Others were to keep ahead of the explosive traffic growth and number of networks served. This included implementing a T3 based network about half way through the five year agreement."

International growth

"The surprise for me has been the dramatically broadened acceptance of the Internet worldwide. One measures for this is the NSFNET backbone traffic increase from about 200 million packets per month in mid 1987 to over a billion a *day* now."

continued on next page

NSFNET Interviews *(continued)*

"Traffic continues to compound at over 10% per month. A second measure is the networks served, about 175 in mid 1987 to over 12,000 now over a third of which are non-US. This demonstrates the International participation in the Internet."

What have we learned?

"We've learned there is considerable value to the educational and research community (and now to others too) when a quality networking infrastructure is in place. It's important to note that this infrastructure is much more than the NSFNET backbone, it includes the good work done by the regional networks and the campus and other local networking support."

What technical advances did NSFNET create or stimulate?

"I've felt that NSF's commitment to TCP/IP in 1987 provided a clearly stated objective to foster the growth we've seen. A lot of technology has been developed and sold in support of TCP/IP routing."

What interesting NSFNET memories can you share?

University–industry collaboration

"There are a set of memories about our NSFNET backbone project that relate to the very positive commitment among our project partnership to do the very best job we could do in providing a quality backbone service. One example of this noted by others is the following quote from the panel NSF selected to do the project's mid-term review:"

> 'One of the most important successes of the NSFNET backbone project has been the demonstration that universities and major high-tech corporations can combine and collaborate in running a large production facility and do it so well. It is worth emphasizing this, since the popular wisdom holds that: (a) universities cannot run a production operation, and (b) that university–industry collaborations are only useful for advanced research projects. What the Merit–IBM–MCI team has demonstrated is that a collaboration between academia and industry is very useful in setting up a production system involving innovative management and new technologies.'

In summary, the NSFNET Project is a dynamic networking environment that is fully responsive to the ever-changing needs of the national community. The Merit–IBM–MCI–State of Michigan collaboration and the associated research initiatives continue to be an excellent model for academic, government and industrial cooperative ventures.

What are the next challenges?

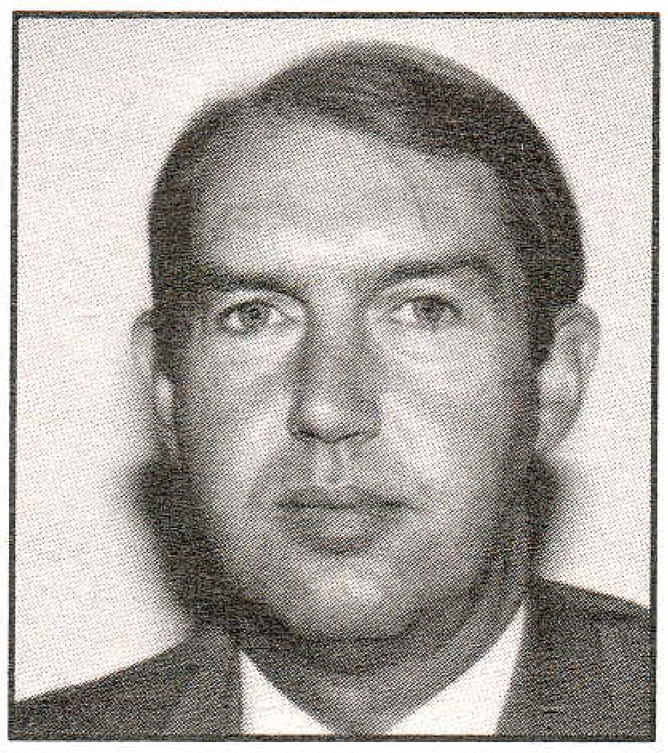
"The next challenge will be dealing with the transition to the new NSFNET/NREN service in a way that is responsive to the needs and growing expectations of the ever expanding user community."

Where do you see NSFNET, NREN and the National Information Infrastructure in 3–5 years?

The new model

"The future is difficult to predict given the uncertainty of the makeup of the NSFNET/NREN service providers and how well the new model for it will work, given the contentiousness within the telecommunications industry and between it and others such as the cable industry, and given the evolving role of the federal government. As a nation we have the opportunity for continuing to have a premier national network and there are many people trying to insure that will be the outcome. There are others seeking to promote their own interests."

Eric Aupperle is currently President of Merit Network, Inc. Merit is a non-profit corporation founded in Michigan in 1966. Merit focuses on educational and research networking in Michigan and through various contracts, regional and national networking projects.



• Dennis Jennings (`jennings%irlearn.ucd.ie@sdsc.edu`)

What was your role in the NSFNET T1/T3 project?

"I was Program Director for Networking at the National Science Foundation from the start-up until the interim 56kbps NSFNET network was implemented. With the support and advice of the network community, we designed and implemented the NSFNET. The choice of TCP/IP (not the undisputed choice at the time), the universal approach (e.g., not just for supercomputer access), the use of routers (instead of switches), the three level organization (campus, regional, and backbone) kept me busy in this position."

What were the biggest challenges and surprises?

"This is a difficult question. Challenges include convincing the supercomputing scientists that universality was in their long term best interests. Finding the appropriate compromises to keep everyone on board and happy with the overall approach. Persuading engineers and high energy physicists that DECnet or MFENET might not be the best solution. Finding a router that could do the job. That the Internet architecture was predicated on the assumption that there would only be a single backbone—ARPANET. Also, discovering that the TCP/IP market was very immature with little software and routers that didn't quite make it (in 1985/86)."

What have we learned?

"That we were naive! I don't think we would have attempted to do what we did if we had realized the technical and other difficulties."

What technical advances did NSFNET stimulate?

The TCP/IP marketplace

"The NSFNET stimulated the TCP/IP market to an enormous extent, and all the subsequent technical developments. If the US Government admitted that it had an industrial policy it would point to the NSFNET as an outstanding achievement."

What interesting NSFNET memories can you share?

"See Carl Malamud's book—*Exploring the Internet.* It was enormously fun and exciting. The incident that I think of at the moment was when I left the NSF and went for a few months as (Interim) President of the Consortium for Scientific Computing and worked with Jeff Schiller on my hands and knees at MIT watching Jeff discover the high voltage on the AT&T T1 circuits! We successfully got the first T1 TCP/IP links working."

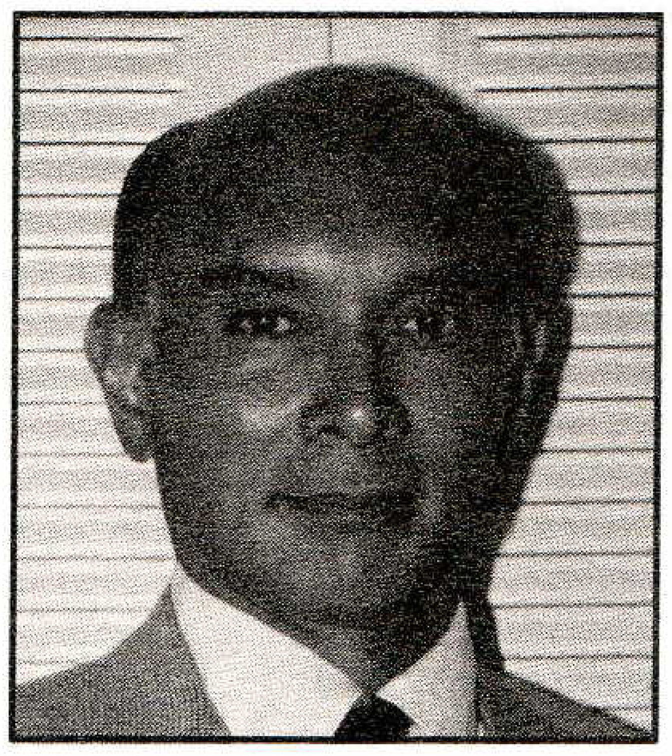
What are the next challenges?

Mission statement

"Realizing that the IP networking market is here and that the days of the NSFNET are over, and making the transition without the whole NSFNET collapsing. Realizing that AUPs are a distraction. Putting more effort into networking services and being as successful at this level as at the lower layers. Realizing that smart use of bandwidth is more important that adding bandwidth. Clarifying the mission—everyone is climbing on board and the mission statement is becoming very confused."

Jennings is Director, Computing Services, University College Dublin.

continued on next page



NSFNET Interviews *(continued)*

- Anthony Villasenor (`villasen@nsipo.arc.nasa.gov`)

What was your role on the NSFNET T1 / T3 project?

"I've been involved with NSFNET since its inception, both as mission agency user of NSFNET as well as a federal colleague of NSF's Steve Wolff and staff."

What were the biggest challenges and surprises?

"When the T1/T3 NSFNET project began, I was quite concerned about negative impacts to the operational stability of the U.S. Internet. I had frightful visions of invaders storming into a big festive party to yank out all the rugs while trying to leave everyone and all the furniture fully intact! But I quickly became a believer in the abilities of the NSFNET consortium to make everything work. There were some [T3] problems to be sure, but the Merit team worked hard and things gradually became stable on the T3 system, and the T1 network was turned off only after the T3 network was proven solid."

Great staff

"The fact that NSFNET could so efficiently execute such a dramatic transition deep within our fundamental network fabric proves that big changes are indeed feasible and possible, given clear focused leadership and an operations-oriented mentality—and that's a valuable lesson to be learned. We were quite fortunate to have the Merit staff working on the design, engineering, testing, deployment and operations of the system—they've been super! And without the NSFNET team pulling all the pieces together, the results could have been far worse. Thus I feel quite optimistic about our collective capabilities to move on to emerging technologies like ATM/SONET, because there'll be so many good people working on it."

What interesting NSFNET memories can you share?

"Memories? Everything about NSFNET is memorable! Beginning with the transition from ARPANET, the spawning of Regionals, the proliferation of research and commercial networks abroad as well as in the U.S., the explosion of network products and applications, the NSF SuperNIC, and even now in the 1994 NSFNET Solicitation, etc. All serve as testimony to the visionary role NSFNET has played and is continuing to play. NSF's leadership here has spawned a whole new industry, and is a fine example of how the government can play a positive role in technology development and national infrastructural support."

Where do you see NSFNET, NREN and the NII in 3–5 Years?

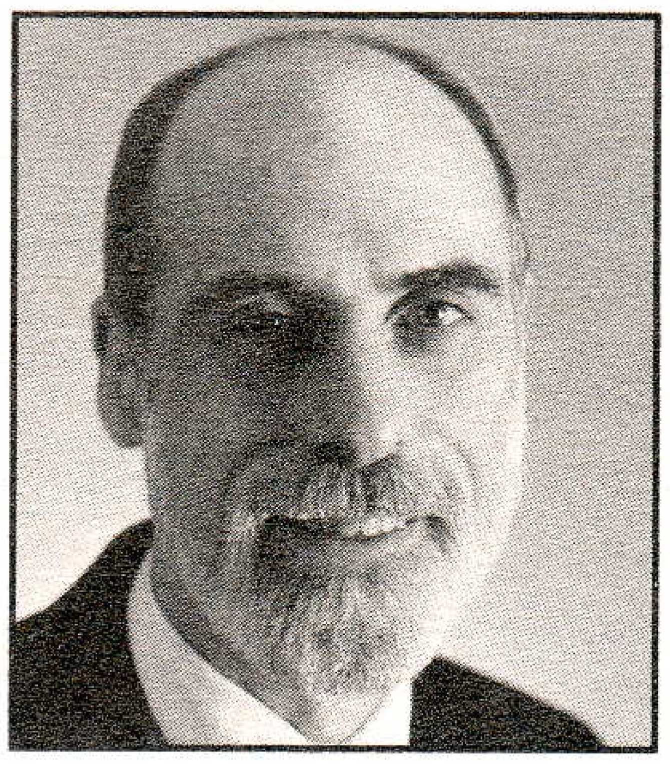
Increased functionality and performance

"Well, given how quickly these new technologies are being developed and deployed, then enhanced by the Internet community's collective genius for creative products and applications, it's really hard to envision exactly what things will look like in 5 years. Certainly, we will be closer to the vision of a ubiquitous BISDN national infrastructure. New capabilities will be provided through increased functionality and performance of the new technologies being designed now. Orchestrating all this will take serious collaboration among government, academia and industry, and there will probably not be enough resources to do everything we want. Of course there will be change, but with the proper focus, we can be even more successful than we've been so far."

A challenging road ahead

"The results will yield considerable benefits for the nation's science, research and education communities, and at the same time increase our nation's commercial competitiveness worldwide. No question about it, the road getting there will be very challenging and very exciting—and lots of fun for all of us!"

Villasenor is Manager, NASA Science Networking (NSI & NREN).



- Vinton G. Cerf (`vcerf@CNRI.Reston.VA.US`)

What was your role in the NSFNET T1 / T3 project?

"I served as chairman of the Internet Architecture Board during much of the period when this activity was underway. I was a member of the IAB throughout this effort. I also served on the Federal Networking Council's Advisory Committee during this period when key policy and technical questions were debated."

What were the biggest challenges and surprises?

"Getting a T3 system to operate was no simple matter. Dealing with the exponentially growing routing databases was equally difficult, as was the problem of coping with appropriate use policies and potential routing loops when maintaining the route information filtering database at Merit."

"The Internet is not only an experimental, research and educational infrastructure. It is also becoming an infrastructure for a wide range of users—most of who simply expect the system to work 24 hours per day."

What technical advances did NSFNET create or stimulate?

Routing developments

"The first T3 wide area packet network was built; and the routing arbiter database concept and many contributions in the form of new routing methods (e.g., border gateway protocol) emerged from the NSFNET effort."

What are the next challenges?

"The transformation of NSFNET into the NAP/routing arbiter structure proposed by NSF is a major modification of the present three-layer hierarchy and will challenge the implementors to maintain high quality service while making the transition. A bit like changing the engines of a jet plane while it is in flight."

Where do you see NSFNET, NREN and the National Information Infrastructure in 3–5 years?

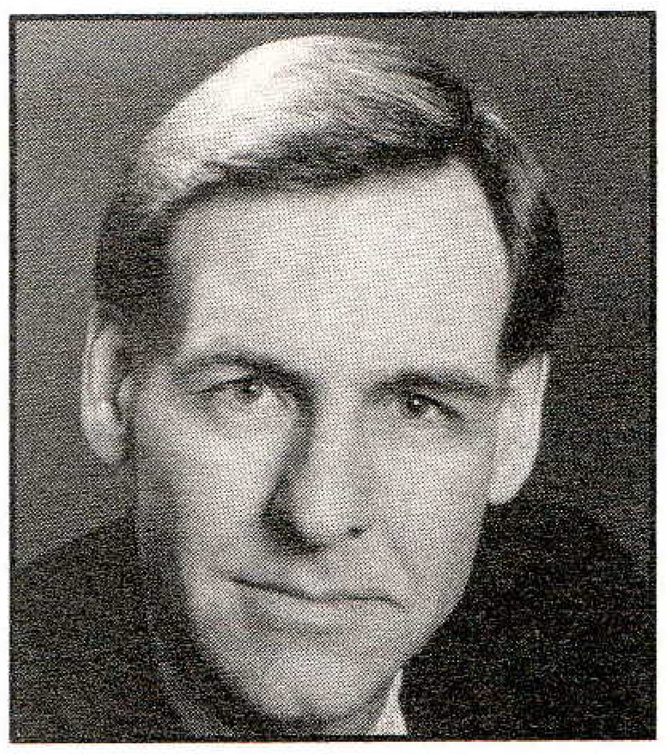
New services

"I expect it will have continued its exponential growth, become a tool of critical importance in the business sector (and for research, education and government), increased its global reach to well over 100 countries, adapted to provide high quality packet voice/video support through multicasting and resource management methods, and gained a permanent place in the national and global information fabric. It will be mostly provided on a private-sector basis, but it will be linked to highly experimental systems that are the focus of government information infrastructure research."

Vinton Cerf is Vice President at the Corporation for National Research Initiatives and President of the Internet Society.

"NSFNET itself may fade as the NAP structure takes hold, but the Internet formed from the US and other national components will persist. It is notable that the three networks on which the Internet was first built (ARPANET, Packet Radio Net and Packet Satellite Net) are all retired now, but the Internet continues to grow."

continued on next page



NSFNET Interviews *(continued)*

• Douglas E. Van Houweling (`douglas.e.vanhouweling@um.cc.umich.edu`)

What was your role in the NSFNET T1 / T3 project?

"I was Chair of the Merit Board at the project's inception, and brought the NSFNET partners (Merit, IBM, and MCI) together to prepare the proposal. I provided strategic guidance on the proposal's content, and wrote some of it. When Advanced Network and Services (ANS) was established, I became Chairman of the Board."

What were the biggest challenges and surprises?

"Bringing the partners together at the outset, working with the networking community to respond to their needs, and coping with the reactions of those critical of the fashion in which the work was being done."

"Also, the difficulty of implementing the T3 network, and the networking community's fractiousness."

What have we learned?

Common goals

"Bandwidth-agile, universally connected networking is fundamental to knowledge workers, and they have been willing, through their enterprises, to invest $100's of millions in such networking. A relatively small investment, such as was made by the NSF on NSFNET, can have enormous leverage. Technology is important, but clear communication and coordination towards common goals are even more critical."

What technical advances did NSFNET create?

"NSFNET's success created a new industry in high speed packet switching—the hardware and software required to implement the network and support its explosive growth."

What interesting NSFNET memories can you share?

Cooperation

"I'll never forget the incredible energy and cooperation that marked the combined Merit/IBM/MCI team as they wrote the proposal for the NSFNET backbone. The work was around the clock, the spirits were high, and the conflicts were intense and honest. I have never seen better teamwork, and that same spirit enabled the partners to implement the T1 network from scratch within budget, on time, and to deliver reliable service from the beginning."

What are the next challenges?

"Making the transition from an already decentralized network management and operations strategy to one that is even less centrally funded and coordinated. Upgrading the protocols and services to meet the rapid growth of user and new applications. Continuing to provide an open network connectivity environment as an increasingly large number of new providers introduce new standards and protocols for services like home shopping, video delivery, etc."

Where do you see NSFNET, NREN and the National Information Infrastructure in 3–5 years?

Doug is currently Vice Provost for Information Technology at the University of Michigan. His responsibilities include the University's computing and telecommunications infrastructure.

"I believe the head start provided by NSFNET gives us the opportunity to continue to provide world leadership in information infrastructure, and that leadership can provide extraordinary benefits to all. Our challenge will be to focus sufficiently on the cooperation among government, education, and industry that has brought us this far, and to not let competition overwhelm the larger strategic effort."

• Jordan Becker (`becker@ans.net`)

What was your role on the NSFNET T1/T3 project?

"I began working on the NSFNET project in 1987 with the IBM group that partnered with Merit and MCI. My initial role was planning, coordination and management of the IBM team that developed the routing and network management software for the T1 NSFNET routers deployed during the summer of 1988."

"Later in 1990, I left IBM to help form Advanced Network and Services, Inc. (ANS), a non-profit company dedicated to expanding the network for the research and education community. In 1991, ANS CO+RE Systems Inc. was established as a wholly owned subsidiary of ANS to provide commercial Internet services. ANS has grown to nearly 100 employees working in several offices across the country. ANS and ANS CO+RE support hundreds of national and multi-national customer organizations that subscribe to the ANS high performance multi- protocol network services."

"My role in ANS has been to lead the expansion and development of new operations and engineering services, and products that broaden the use of the network service. I have worked closely with many users, and midlevel subscribers of NSFNET services to manage the growth of the service."

What were the biggest challenges?

Growth in two dimensions

"The biggest challenge has been managing the explosive growth of the NSFNET. The growth has occurred in two dimensions. One dimension is traffic which continues to grow at an exponential rate. The other dimension is routing complexity. NSFNET provides transit backbone connectivity between midlevel, regional, and campus networks in the U.S. and overseas. There are over 10,000 destinations that can be reachable in real time under ANS management. The growth in reachable destinations has also accelerated at an exponential rate. The development of new performance enhancements, protocols, and management tools has continued at a rapid pace since work began on the project in 1987."

"Managing this growth has been difficult at times. It is very hard to engineer a system that grows at an exponential rate. There have always been considerable resources provided to support the project by the partners, however we have frequently had to make decisions to trade off longer term enhancements and features for near-term solutions to scaling problems that avoid hitting the edges of the performance/growth envelope."

What were the biggest surprises?

Information services

"There were several big surprises. One has been the ways in which the network has been used when considerable capacity became available. The emergence of new information services tools like *Gopher, Archie*, and *WAIS* was unexpected. The recent emergence and explosion of packet audio and video application and traffic was also unexpected. NSFNET is a living example of the '*Field of Dreams* Phenomenon' ["build it and they will come"]."

"Another big surprise was the emergence of a thriving commercial Internet service. This was facilitated by the NSFNET service since many of the early commercial Internet providers started as subscribers of the NSFNET service."

continued on next page

NSFNET Interviews *(continued)*

What have we learned?

"We have learned quite a lot on the NSFNET project. We have learned about networking technology including routing protocols for large systems, network management tools, data traffic accounting and performance analysis, high bandwidth applications development, and distributed operations and coordination among many decentralized autonomous service providers."

What technical advances did NSFNET create or stimulate?

Proving ground

"NSFNET has been a proving ground for new technologies that have been conceived out of practical necessity, and trialed under fire. These technologies have included development and deployment of new routing protocols like the *Border Gateway Protocol* (BGP), and the *Intermediate System–Intermediate System* (IS–IS) protocol. SNMP network management protocols and tools where exploited early on the NSFNET. The first policy routing database tools for global routing coordination was pioneered by Merit on the NSFNET project. The first high bandwidth video and graphics demonstrations were established on the early T3 NSFNET system."

What interesting NSFNET memories, events or experiences can you share?

Talented people

"Most of my memorable events involve working with the collection of extremely talented people that have contributed to this project over the years. Just a few of these memories include:

- Working with Jeff Case and Ken Key on the early SGMP agent software before deployment of the T1 NSFNET (March 1988). Maintenance on the base code was paid for with six-packs of Diet Coke.
- Hans-Werner Braun leading the turn-up of production service on the T1 NSFNET backbone right on time (July 1, 1988) only 8 months after the award was made.
- Surpassing the 100M-packet and 1000-network month.
- Yakov Rekhter deploying the first BGP implementation on the T1 NSFNET.
- OSI CLNP switching services deployed on the T1 NSFNET.
- Dave Katz developing the first NSFNET FDDI driver, and testing it at INTEROP. This is first of several Dave Katz FDDI drivers.
- The first wide-area T3 demonstration at the National Net '90 conference in Washington D.C. (March 1990). The hotel air-conditioning was not sufficient for high-speed router and computer equipment, so Steve Heimlich improvised by using an "Ice-Swan and Fan" combination. [The room was cooled by the air blown over the large ice carvings in the shape of swans.]
- Deployment of the first nationwide public T3 NSFNET service (December 1990).
- Turning off the T1 NSFNET backbone on December 4th 1992.
- Surpassing the first 1 billion packet day on ANSNET/NSFNET in early 1993."

What are the next challenges?

"Managing the explosive growth of the NSFNET routing table is an immediate challenge. New software will increase the forwarding table capacity to 25K destinations, and will include support for the new *Class-less Inter-domain Routing* (CIDR) features to slow the growth in Internet routing complexity."

"The T3 NSFNET service will be enhanced with higher performance equipment and redundant switching systems."

"Native OSI switching services will be supported on the T3 NSFNET service."

Where do you see NSFNET, NREN and the National Information Infrastructure in 3–5 years?

More growth

"The explosion of commercial public and private Internet connectivity will continue. Research and education applications will begin to leverage gigabit speeds available on the future network. The network will become more pervasive in reach and application, extending well beyond the current base of 10 million users."

Jordan Becker is the Vice President for Network Services at ANS, responsible for the engineering and operation of the ANSnet. The ANSnet network supports NSFNET backbone services.

References

[1] Dern, D., "The ARPANET is Twenty," *ConneXions,* Volume 3, No. 10, October 1989.

[2] NNSC Staff, "Profile: NSFNET," *ConneXions,* Volume 1, No. 2, June 1987.

[3] Braun, H-W., "The new NSFNET backbone network," *ConneXions,* Volume 2, No. 12, December 1988.

[4] Roberts, M., "The NREN Takes Shape," *ConneXions,* Volume 5, No. 6, June 1991.

[5] Roberts, M., "NREN Bill Signed into Law," *ConneXions,* Volume 6, No. 2, February 1992.

[6] Roberts, M., "NREN Politics Thicken," *ConneXions,* Volume 7, No. 1, January 1993.

[7] Roberts, M., "The NREN Moves Towards a National Information Infrastructure," *ConneXions,* Volume 7, No. 8, August 1993.

[8] Cerf, V., "A National Information Infrastructure," Written Testimony for the US House of Representatives Committee on Science, Space and Technology Subcommittee on Technology, Environment and Aviation, March 23, 1993, *ConneXions,* Volume 7, No. 6, June 1993.

[9] Adams, A., "The Merit Policy-Routing Configuration System," *ConneXions,* Volume 7, No. 2, February 1993.

[10] Deutsch, P. "The Internet as Service Provider," *ConneXions,* Volume 7, No. 9, Septemkber 1993.

RICHARD JAY SOLOMON is a Research Associate with the MIT Program on Digital Open High Resolution Systems (DOHRS) in the Center for Technology, Policy and Industrial Development, and at the MIT Research Lab on Electronics. He is a policy analyst specializing in telecommunications technology. He has also assembled a history of computer and networking pioneers in 90 hours of video taped interviews. E-mail: `rjs@farnsworth.mit.edu`

Making Standards the IETF Way

by David Crocker, Silicon Graphics, Inc.

Balancing technology and procedure

The Internet began as a research activity by the U.S. Defense *Advanced Research Project Agency* (DARPA) and has developed into a global data communications service, operated as a loose confederation of many different organizations. At the core of this service is a collection of networking technologies that were originated by the DARPA-funded researchers but which now benefit from improvements and additions by an equally loose international confederation from research, academia and industry. The Internet currently is estimated to include about 10,000 networks and 1–2 million hosts (and maybe as many as 15 million end-users). It is doubling approximately every year, so its technology is reaching further into the general population.

Given the variety of other activities and groups pursuing development of communication standards, the success of the Internet and its technology is remarkable. This article discusses the style of technical development that is used within the Internet and suggests the reasons for its success. Some comparisons with other standards effort are offered, as well as an attempt to gaze into the future for the Internet's technology development. An extensive discussion of this topic can also be found in [1]. The formal description of the Internet standards process is documented in [2].

However, it is first useful to discuss the realm of standardization in which the Internet developers play. Called "open systems" the term has come to have very different meanings.

What's in a standard?

In data communications, a standard specifies a set of procedures. Typically pertaining to computer-to-computer interaction, a specification might be more limited, such as describing only the format of data, rather than all of the rules for passing that data back and forth. While mildly controversial, it also is legitimate to specify characteristics of information that is exchanged among *humans,* such as for electronic mail address strings placed on business cards. Standardizing such strings greatly facilitates the "out of band" passing of information which eventually winds up as input data to a computer.

A standard might also specify the procedures to use when operating a system. Typically, Internet standards shy away from such dictates, since there is a strong desire to leave network operators free to conduct business as they see fit. However, guidelines occasionally are published, when conformance to them will be highly beneficial for the overall health of the Internet. Still, such guidelines are not formal standards.

Who adopts a standard?

Discussions often distinguish *de jure* from *de facto* standards. As the name denotes, the former is made legitimate by force of law, whereas the latter is legitimate by virtue of popularity. Since the Internet's researchers had no intention of developing a global service, its technology definitely falls into the camp of *de facto.* Unfortunately, this is sometimes used against it, to suggest that it is less legitimate than the formally-commissioned products of other groups. Instead, one should note that its adoption has been possible only because of its very strong virtues.

By and large, *all* successful specifications are *de facto* standards, since there is little leverage that their developers have on those doing the adopting. *De jure* standards can, and do, fail to gain popular use. So, it's much more helpful to consider these technologies on their merits than on their pedigrees.

What does "open" mean?

• *Open publication:* The commercial pressure for open systems has been specifically intended to let customers obtain products from a variety of vendors, potentially buying each component with a competitive bid. But there are different ways to create multiple sources of a product, so the remainder of this section considers the options and particularly the types of organizations that produce these various kinds of open systems.

It is possible for a vendor to publish the specifications of their proprietary technology. This allows a third-party "aftermarket" to exist, usually selling products at a lower price than the vendor who owns the specification. At any time, however, the vendor may choose to change the specification and delay publication of the changes until after the vendor has released their own new products.

Another concern is that specifications are not universally available. For example, requiring consortium membership with high membership fees, effectively restricts the free flow of information from the community at large. Certainly consortia have special advantage by controlling the content of a specification, while preventing community-wide review of its choices.

• *Open ownership:* Traditional, "accredited" standards bodies have relatively liberal rules of membership and conduct open meetings. They publish their specifications, though usually for a significant price, making them available to any customer or vendor. No single company and no market-driven consortium control the specifications, allowing vendors to work from a reasonably level playing field. Work is done only at meetings which are held at venues around the world. This requires major investment by anyone wishing to attend, constituting an implicit barrier to broad-based participation.

• *Open development:* The most extreme approach develops specifications in a forum open to anyone who is interested in participating, allowing on-line contribution so that travel is not required. The results then are also available to all, at little or no charge and in a highly convenient on-line format to anyone interested in reading them.

Selection of technical topics also can be by open process. If a topic lacks an adequate constituency, it's not pursued. If a topic has diverse constituencies they are free to go their own ways and the market chooses among them. Continuing on-line discussions, away from the meetings, allows progress to be made quickly.

Best technology versus reasonable political compromise

A standards development process must perform a difficult juggling act. It must select among a range of technical alternatives, and it must do so in a matter that attends to the political concerns of its members. A process which attends only to technical excellence may produce a solution which is applicable only in a very narrow context. For example, it might not provide an adequate transition path for a large installed base of users of older technology. However, if the process places too much emphasis upon polite accommodation of the desires of each and all its members, the well-known problems of "design by committee" are guaranteed to sabotage the results.

A communication standard always is responding to the needs of several constituencies. At the least, there are product developers, service providers, and end users. Determining their needs is difficult. Accommodating all of those needs usually is impossible.

continued on next page

Making Standards the IETF Way *(continued)*

The IETF standards process

From a small effort from a few researchers, the Internet's technical development effort has grown considerably. Today's work is done by a group known as *The Internet Engineering Task Force* (IETF). In 1987, its attendance numbered 40 souls. Today, approximately 700 people attend its thrice-annual week-long working meetings.

History

The original ARPANET effort involved very focused research on basic issues of packet switching. However, much of the use of the technology was subject to development by happenstance. The informality of the process had the detriment of relying entirely upon the energies of one or a few "champions" rather than the more deliberated outcome of an organizational commitment. Documentation tended to be incomplete at the start and was not revised in a timely fashion. On the other hand, it had the great advantage of being produced quickly while being only part of the shared knowledge needed to produce interoperable systems. The rest came from attending the working group meetings. Another feature of the informality was that a scribe could make "enhancements" to the specification and have them implicitly accepted—if no one objected too loudly. The original ARPANET mail facility was the result of just such a casual, private decision.

Since the community was geographically distributed, but specifications and ideas needed quick dissemination, an on-line publication series called *Request for Comments* (RFC) was initiated in 1969. The name very accurately reflected the desires of authors. RFCs were explicitly viewed as working documents to be used within a relatively small community. They ranged from casual ideas to detailed specifications and from expressions of operations concerns to whimsical fantasy. If an idea seemed attractive, an individual might spontaneously specify a protocol or a group might meet to discuss it further. If a protocol seemed useful, someone implemented it and if the implementation was useful, it was copied to similar systems on the net.

By 1981 the Internet effort, which followed the ARPANET effort, had matured and grown to the point that the DARPA Program Manager decided to form an advisory group, called the *Internet Configuration Control Board* (ICCB) and having the task of giving DARPA technical advice. Initially consisting eight members, this is essentially the management structure that is in place today. In 1984, it was renamed to the *Internet Activities Board* (IAB).

In 1987, the IAB created the *Internet Engineering Task Force* (IETF) and the *Internet Research Task Force* (IRTF). The former was chartered to provide near-term solutions to technical difficulties in Internet operations and to develop near-term enhancements for the Internet. The latter group was tasked with pursuing those topics of long-term interest which carry some technical risk.

Because the bulk of the funding for TCP/IP research and development initially came from the U.S. military establishment, there is a natural tendency to assume that the work was fundamentally biased towards the needs of the United States. However one of the three original research groups to work on TCP/IP was University College London, in England. As the ICCB formed, a body called the *International Collaboration Board* (ICB) was formed at the same time and usually met in parallel with it, often in Europe or Canada. The ICB had a European focus, with the goal of coordinating requirements of transatlantic and NATO use of TCP/IP, particularly in the context of the multi-site *Atlantic Packet Satellite Network* (SATNET), which included Norway, United Kingdom, Italy and Germany.

The success of the Internet and its technology, in particular with its expanding commercial market and international scope, has created pressure for a more formal affiliation. There was some exploration of an association with an existing standards body, but without productive outcome. The result was the January, 1992 formation of a professional organization, called the *Internet Society* (ISOC). In June, 1992, the IAB was placed under the ISOC with responsibility for "...oversight of the architecture of the worldwide multiprotocol Internet" including continued standards and publication efforts. As part of the move, the IAB changed its name to be *Internet Architecture Board,* since the IAB does not, in fact, participate directly in the operational activities of any Internet component.

Organizational structure

The ISOC and IAB bodies serve to provide ultimate oversight to the IETF standards efforts. Direct, line-management of the process comes from the *Internet Engineering Steering Group* (IESG), which charters efforts and approves their results. The IESG comprises the IETF Chair, and a number of *Area Directors* who oversee efforts within various technical areas. The list of areas changes from time to time, but the current list is:

- User services
- Applications
- Service applications
- Transport services
- Internet services
- Routing and addressing
- Security
- Network management
- Operations requirements
- Standards management

OSI-related activities were managed by a separate OSI Integration area. The IESG recently decided that such protocol work was now broadly and fully incorporated into the full range of IETF activities.[3]

The productive efforts of the IETF are performed in *working groups.* Each working group has a chair. Anyone who participates in working group activities, on-line or at meetings, is considered a member. The real challenge to working group management is balancing the requirement to give full and fair hearing to all sides in a debate, while still ensuring that forward progress is made in reaching the working group's goals. As membership has become larger and more diverse, many working groups find it difficult to develop specifications entirely within working group plenary sessions (face-to-face or in on-line discussions). As a result, homogeneous, self-selecting groups, called *Design Teams* have formed. They conduct the core of the specification work, responding to requirements and suggestions made by the working group. While there is occasional concern about the leverage that a Design Team can have over the contents of a specification, the team always is subject to the consensus of the working group.

Working groups are commissioned with a charter that details goals and schedule, either of which may be renegotiated as the working group progresses. A typical working group operates for 9–18 months. When it produces a specification that formally enters the standards track, the working group goes quiescent, although its mailing list remains operational and often is quite active.

continued on next page

Making Standards the IETF Way *(continued)*

A small *IETF Secretariat* provides the substantial support effort needed to mount a major, week-long meeting three times a year, run many IESG and IAB teleconferences, and otherwise perform the administrative legwork of this volunteer organization. The secretariat is administered by the Corporation for National Research Initiatives, with funding from several U.S. government agencies. Over time, support is expected to come from a broader base of international private and public organizations.

Working documents of the IETF are maintained as part of a replicated, on-line store, called the *Internet Repository*. Documents under development are called *Internet Drafts* (ID). Documents which have reached a level of stability, possibly by attaining a standards track status, are published in the continuing *Request for Comments* (RFC) series. Those that have a standards status also are assigned an *STD* number. Protocol specifications permit a wide range of enhancements to be registered and the issuance of registered values for these is provided by the *Internet Assigned Number Authority* (IANA). RFC publication and IANA operation are from USC's Information Sciences Institute.

Standards labels

A document intended to be an Internet Standard goes through four stages of maturity. The first is basic development, during which time the specification has no formal status and might not result in a submission to the standards process. When the specification is stable, has a sufficient constituency, and has no known omissions or problems, it may formally enter the standards track as a *Proposed Standard*. In general, testing before standardization is an important principle of the Internet process. Although implementation and testing are not required in all cases before entering the standards track, they generally are encouraged.

A specification may be submitted for elevation to *Draft Standard* when there exist at least two independent implementations which have interoperated to test all functions, and the specification has been a Proposed Standard for at least six months. When a Draft Standard has gained significant field experience, providing a clear demonstration of community interest in using the specification and has held its status for at least four additional months it may be elevated to the status of full *Internet Standard.*

Documents which are produced by other standards bodies, other organizations, or individuals simply wishing to make their work available to the Internet may publish a version as an RFC, with a status of *Informational.* These are *not* Internet standards and are not intended to be the subject of direct Internet effort. Specifications which are not on the standards track but which the author seeks to gain Internet experience may be published as *Experimental.* The specifications may change, may be incomplete in some respects, or may contain significant errors. However, the specification's author wishes to encourage technical review and experience, possibly for later consideration in the standards process.

Most work is in the production of *Technical Specifications* (TS). These are the familiar descriptions of formats and procedures. However, there may be separate *Applicability Statements* (AS) which describe the circumstances under which one, or more, TSs are to be used.

Acceptance criteria

When considering a specification for adoption as an IETF standard, the general criteria that are applied are:

- *Competence:* The specification is technically sound and consistent with the overall Internet architecture.
- *Constituency:* There has to be a significant set of potential providers and of potential users and an indication that they will, in fact, use the services provided by the specification.
- *Coherence:* The specification must be written clearly and cleanly.
- *Consensus:* The specification must reflect an adequate consensus of the technical community.

Standards procedures

The process for creating IETF standards is relatively simple, although it has become more formal over time. A working group is chartered and a working group chair is assigned by the IESG. The working group then conducts its business on-line and at IETF meetings. (Additional meetings are allowed, but are relatively rare.) Each working group establishes its own details for operations, ranging from a loose, conversational style, to much more formal and structured attacks on well-defined problems.

At base, the keys to a working group's operation is the requirement that it reach a "rough" consensus about decisions and that it make those decisions in a manner which achieves forward progress towards the goals stated in the charter. When the working group agrees that it has a stable specification which satisfies appropriate technical requirements, it submits it to the IESG for approval. A brief, public review permits final expression and evaluation of concerns about technical content or working group process.

Conflict resolution

There is no voting in working groups, since there is no formal membership. This guarantees moments of divisiveness, since parties that lose various debates will occasionally feel that they were not given a fair opportunity to express their views or that the consensus of the working group was not accurately read. All such expressions of concern are taken very seriously by the IETF management. More than most, this is a system that operates on an underlying sense of the good will and integrity of its participants. Often, claims of "undue" process will cause a brief delay in the standard-track progression of a specification, while a review is conducted. While frustrating to those who did the work of technical development, these delays usually measure a small number of weeks and are vital to ensuring that the process which developed the specification were fair.

The salient points in IETF success

The Internet standards process never set out to achieve its current role. It was only the side-effect of a small research community. While that community was reasonably clear about the basis for its good work, the global perception of that success is quite recent. Hence, it is worth considering what constituents go into this remarkable process.

- *People, not Procedure:* For all of the increasingly formal procedure in the IETF standards process, the real work of the IETF relies on individual judgment, as well as individual effort. The formal rules provide beacons for guidance and for synchronization.

The real test that is applied to difficult choices is whether the people involved conducted themselves fairly and made the best choices under the circumstances. Reliance on general "rough" working group consensus is the constant check-and-balance to potentially misguided behavior of individuals.

continued on next page

Making Standards the IETF Way *(continued)*

• *Pursuit of core vision:* Usually, an IETF specification is the clear and direct result of a specific technical vision. One, or a few, individuals see a solution and recruit others to that vision. While the working group's membership can, and do, dictate changes, successful working groups are careful to maintain the integrity of the original vision.

• *Simple, immediate goals:* IETF specifications usually attempt to solve specific, immediate problems, rather than to encompass a wide-range of long-term goals. This permits work to be directly responsive to immediate requirements. Keeping goals simple tends to make the resulting designs also simple. While some might call the results "limited," others call them "elegant." Typically they do prove to be quite extensible.

• *Incremental enhancement:* Because it can field solutions quickly, Internet standards can benefit from considerable operational feedback. This, in turn, permits another round of specification, *if needed.* While too much iteration would certainly result in unstable specifications, this problem happens rarely.

• *Diverse contribution:* Newcomers to the IETF never quite believe that the process is as open as it is. Anyone with a fresh perspective, clear insight, or good jokes is always welcome. As the Internet, itself, increases its global reach, many IETF contributors participate exclusively by e-mail. While attendance at IETF meetings is extremely helpful, it is not required to be an effective working group member. Since it is easy to join a working group mailing list, many members remain silent until some aspect of debate triggers their interest or calls on their special expertise.

• *On-line collaboration:* As described above, the ability to have on-line working group participation is paramount. It fundamentally eliminates the barriers of time and cost for membership and contribution. This increases the number and diversity of people who can contribute enormously. Further, it means that progress does not have to wait for the next meeting.

• *Comprehensible specifications:* The IETF has very loose requirements for the style in which its standards are written. In general, this results in documents which are easily read by the average implementor. Although formal analysis often uncovers ambiguities and errors in such documents, the informal network of implementors convey whatever additional information is necessary. This is certainly not an ideal system, but seems to balance flexibility and immediacy well enough to be highly productive.

• *Easy access to the specifications:* The Internet Repository means that anyone with Internet access can obtain standards and working documents of the IETF, for *no additional cost.* This is in marked contrast with the standards organizations. It is another example of the ways in which the IETF work is highly accessible to the broadest audience, permitting better analysis and broader use.

• *Live with the results:* IETF participants usually are directly involved in producing or using the technology. In particular, they are not professionals in standards development. Even more important, IETF members build what they specify and then use it. The Internet, itself, provides a very large scale live test environment and as is often true with software, once it passes the tests it is instantly used in production. If a working group's efforts are not useful, this is quickly evident.

Comparison with typical standardization efforts

It can be quite telling to look at the real membership requirements for organizations which declare themselves in the business of developing "open" specifications. They usually have very severe membership filters, in terms of membership cost or travel expenses needed to go the meetings. These expenses usually seem small, for most businesses. But the costs often serve to exclude smaller businesses, various research and education organizations, and personal participation by those without an appropriate organizational affiliation. This necessarily restricts the range of views that can be offered to the development process.

Technical goals

Most standards efforts seek to solve a problem in the most general manner and for the longest-term possible. Such intentions cannot be challenged. They are well-meant. Unfortunately, the goal of extreme generality requires very long and careful analysis and requires attending to a very broad range of requirements, which further adds to the design and analysis burden. Hence, it usually takes a very long time to produce these general solutions. Hence, these solutions often have missed their window of opportunity. Worse, they often have become cumbersome, difficult to implement, resulting in very large software modules.

IETF work occasionally suffers from these problems, too. In fact, the IETF is not very successful at fixing working groups that make the mistake of walking down the seductive path of long-term, general design. Fortunately, most IETF working groups operate with narrow vision, trying to solve immediate problems. The wide range of views that contribute to the work usually make painfully clear what features a specification is lacking. As a result, designs often include hooks for later extensions, so that those who did not get their favorite feature into the current draft, can *separately* specify enhancements. If the community decides that one or another enhancement is valuable, it gets adopted. But the evaluation process for the additional features does not impede development and adoption of the functional core.

By rights, the narrow focus and near-term goals of the IETF work should makes its specifications rigid and short-lived. Real-world experience shows a difference performance record. The specifications are comprehensible to a broad range of implementors. The software operates on the complete range of platforms and is useful in most data communications contexts. Better still, its utility continues after more than ten years of production use.

As the Internet technology is applied to a wider range of environments, various deficiencies are identified. Security and accounting are the ones most commonly cited, though support for guaranteed levels of service, such as for real-time traffic, also are noted. To date, the IETF has shown an astonishing ability to add capabilities to the core technology. To date, there is little indication that it has reached a limit in that ability.

The ironic contrast

While the differences in market-acceptance and use of the Internet work, versus that from the OSI community, over the last five years are marked and clear, it's puzzling to try to determine the engineering rule of thumb that would explain it. Here is one attempt:

The OSI community's desire for functional completeness and accommodation of all interests leads to the philosophy of *including* as much as possible in a design. In contrast, successful IETF working groups are driven by near-term needs and consequently try to produce designs that *remove* as much as possible.

continued on next page

Making Standards the IETF Way *(continued)*

At first blush, this should produce highly limited designs. The trick in the process appears to be the group consensus requirement. As one would expect each participant contributes their list of desired features, but the short time-fuse on the work requires that the group reach consensus quickly. This can only be done by removing features, since only a small core of features will be clearly acceptable to most participants. (The latter approach of including all of everyone's preference requires too much group debate and results in a design that is too-obviously unacceptable.) However, the process of removing features also requires some assurance that some of those features can be added later. Hence, the design usually permits extensibility which is itself, designed with an approximate sense of the sorts of extensions that are likely to be made.

The future of IETF standardization

Surely there is a down-side to the good-will and good results of the IETF? And indeed there is. The IETF's growth is proving a fundamental challenge to its style of operation. More people mean less familiarity on a personal and professional level. Internet technology now represents a multi-billion dollar business. Hence, IETF decisions have significant financial impact and that can raise the heat of a debate quite a bit.

While more people are participating, the number of senior, experienced contributors has not risen proportionately. Such folks are essential for providing working groups with guidance about successful practice. Without such guidance, working groups run the serious risk of have good consensus about a bad design.

In general, the IETF is applying its own technical design philosophy to its own operation. So far, the technique seems to be working. With luck, it will demonstrate the same analytic likelihood of failure, with the same experiential fact of continued success.

Acknowledgements

Jackie Snell provided patient review and many suggestions for improving the text. Stephen Crocker provided the clear, coherent, concise wording describing the IETF acceptance criteria.

References

[1] Crocker, D., "Evolving the System," in *Internet System Handbook,* edited by Daniel Lynch and Marshall Rose, Addison-Wesley, Reading, Massachusetts, 1993, ISBN 0-201-56741-5.

[2] Chapin, A. L., "The Internet Standards Process," RFC 1310, March 1992.

[3] Huizer, E., "The IETF integrates OSI related work," *ConneXions,* Volume 7, No. 6, June 1993.

Learn more in Session S49: "Internet Standardization," Thursday at 3:30pm.

DAVID H. CROCKER has participated in the development of internetworking capabilities since 1972, first as part of the ARPANET research community and more recently in the commercial sector. He wrote the current Internet standard for electronic mail header formats (RFC 822). Mr. Crocker currently serves as the IETF Area Director for Standards Management, responsible for developing an enhanced application protocol infrastructure. This article is the result of his previous service as Area Director for Standards Management. His recent IETF technical efforts include the IPAE specification for transition to IP The Next Generation, and STIF, a proposal for encoding structured text objects. Mr. Crocker is currently with Silicon Graphics. E-mail: `dcrocker@sgi.com`

 (See also page 95 of this issue.)

An Experiment in Remote Printing

or

Toward the Integration of the Internet and Telephony

by Carl Malamud, Internet Multicasting Service
and
Marshall T. Rose, Dover Beach Consulting, Inc.

Introduction

The Internet is perhaps best described as a general-purpose infrastructure for computer-communications—it supports a variety of applications, which represent a wide range of requirements.

The international telephone network is another infrastructure, although it is used primarily for connecting special-purpose devices such as telephones and G3 facsimile printers. Though data communications environments with general-purpose computers (e.g., private networks and the Internet) build on the services of the underlying telephone infrastructure, the special-purpose devices have remained inaccessible to computer users, effectively fragmenting global communications services into special application domains such as fax and phone.

Earlier this summer, we started an experiment to see how well we could integrate the Internet and telephony infrastructures.

The first application

In order to study this problem, we decided to start by trying to integrate the e-mail and facsimile communities. Many sites cooperatively provide "remote printing" access to the international telephone network. This allows people to send facsimiles via e-mail. The Internet e-mail infrastructure takes care of all the routing, delivering the message to the appropriate remote printer server in a manner totally transparent to the user.

E-mail and facsimile are very similar applications, so this is a good starting place for our research. However, there is another important benefit: we believe that by providing easy access to remote printing recipients, enterprise-wide access is enhanced, regardless of kind of institution (e.g., commercial, educational, or government), or the size of institution (e.g., global, regional, or local). This approach at outreach allows an organization to make it easier for the "outside world" to communicate with personnel in the organization who are users of facsimile but not e-mail, such as the sales person, the university registrar, or the (elected) official.

Addressing

So, what does a facsimile recipient's address look like? Like all Internet e-mail addresses, it has two parts: `local@domain`.

The "local" part contains an indication that you are using the remote printing service, along with the identity of the recipient, e.g.,

```
remote-printer.Arlington_Hewes/Room_403
```

The "domain" part identifies the address of the facsimile device (its telephone number). Start with the "international" version of the telephone number, e.g.,

+1-415-968-2510

remove the punctuation, invert the digits, and place it under a special subdomain called `tpc.int`, e.g.,

```
0.1.5.2.8.6.9.5.1.4.1.tpc.int
```

continued on next page

An Experiment in Remote Printing *(continued)*

Putting it all together, to send to someone named "Arlington Hewes in Room 403" on the fax machine at +1-415-968-2510, you would use:

```
remote-printer.Arlington_Hewes/Room_403@0.1.5.2.8.6.9.5.1.4.1.tpc.in
```

Of course, you probably wouldn't want to type this in—your user agent should give you facility for generating the address automatically!

If an Internet site is providing access to this telephone number, the Internet will automatically route the message to that site where a cover sheet will be generated and sent along with the body of the message. Of course, if you are using *MIME* (the Internet's multi-media mail standard), then in addition to plain text, you could include formatted text, *PostScript,* and images as well.

The `tpc.int` subdomain

The key to providing this remote printing facility is to use the Internets *Domain Name System* (DNS) to provide a mapping between addresses in the telephony world (i.e., telephone numbers), and those Internet sites willing to provide access to the devices at those addresses (e.g., telephones or fax machines).

The basic idea is simple: a participating Internet site registers itself in the DNS for one or more telephone number prefixes. Since both domain names and telephone numbers are hierarchical, it's easy to do this mapping. For example, suppose a company wanted to register itself as providing access to all of its own telephone numbers. It simply takes the prefix(es) corresponding to those numbers, e.g., +1-415-336, removes the punctuation, inverts the digits, and places it under the tpc.int subdomain. For example,

```
*.6.3.3.5.1.4.1.tpc.int    IN MX 0 foo.com.
```

indicates that traffic for any number in this prefix should be directed to the Internet host `foo.com`

There are really four kinds of sites which participate in the `tpc.int` subdomain:

- Neighborhood sites
- Regional sites
- Enterprise sites
- Personal sites

A *neighborhood site* is someone who provides access to any telephone in its "local calling area." The idea being that metered access in this area is fairly inexpensive, and the site is willing to provide access as a part of their community computing environment.

A *regional site* is basically just a large neighborhood site, usually providing access to an entire country or a large part of a country, such as an area code. The interesting thing to note is that regional sites may choose to shrink or expand their cell, depending on factors such as demand and cost.

An *enterprise site* is a company or other organization that provides access solely to its own facsimile machines. They register exactly those telephone prefixes which apply to their enterprise, regardless of whether the organization is located in one area, or is a multi-national.

A *personal site* is someone who provides access to exactly one telephone number, usually one that attaches to their desktop. For remote printing, when the server gets the message, it will usually just deliver it to the owner of the desktop—via e-mail.

CARL MALAMUD is President of the Internet Multicasting Service, a public benefit corporation in Washington, D.C. that operates the Internet Talk Radio and Internet Town Hall channels. The host of "Geek of the Week," Malamud is the author of seven professional reference books including *Exploring the Internet* (Prentice Hall, 1993) and *STACKS* (Prentice Hall, 1992).

MARSHALL T. ROSE is Principal at Dover Beach Consulting, Inc., a California-based computer-communications consultancy. He spends half of his time working with clients, and the other half involved in self-supported, openly-available projects, as a theorist, implementor, and agent provocateur. His subscriptions to *The Atlantic, Rolling Stone Magazine,* and *Wired!* are in good standing.

Policy and economics

One of our interests in the experiment is studying the policy and economic models which apply to the sites participating in the `tpc.int` subdomain. For example, a company might run a neighborhood cell because it's a good Internet citizen. In contrast, some neighborhood cells, and many regional cells, might sell advertising space on the cover sheet (e.g., "This fax brought to you by Blotto Computers—When YOU think computers, think Blotto!") In fact, a cell site might very well be operated as a profit center—which leads to an interesting economic model, where neither the originator nor the recipient are involved in the settlement transaction!

The `tpc.int` subdomain is operated with emphasis on the principles of fairness, security, and observance of all applicable laws. Toward this end, the subdomain is organized as a cooperative, with a council that arbitrates matters on behalf of participating sites. As a part of this, the council oversees policy on issues such as auditing, privacy, and denial of access because of abuse.

Current status

The official kick-off of the experiment was 16 July 1993. By the end of July, service was up and running for:

- All of Australia (provided by a regional cell)
- All of Washington, DC (provided by a regional cell)
- Most of Silicon Valley (provided by some neighborhood cells)
- Parts of Riverside (provided by a neighborhood cells)
- And all of the University of Michigan (an organizational cell)

In addition, we expect the following countries to come online later in the summer of 1993:

- Denmark
- Germany
- Finland
- Ireland
- Japan
- Sweden

along with several enterprises such as companies and government R&D centers.

There is an openly-available implementation of both client and server software for UNIX. (The server software is based on Sam Leffler's *FlexFAX* package.) In addition, InnoSoft's *PMDF* software for OpenVMS also supports remote printing. Naturally, if you maintain a commercial or public e-mail package, we'd like to hear from you.

For further information

At the INTEROP August 93 conference, we will be hosting a Birds of a Feather, in which we'll be discussing both the technical and policy aspects of our experiment. In addition, there's an Internet mailing list on remote printing. Send a note to:

`tpc-rp-request@aarnet.edu.au`

The Remote Printing BOF will be held Wednesday, August 25, at 7:30pm.

and ask to be added. You can also get a copy of a FAQ (frequently asked questions) document by sending a note to:

`tpc-faq@town.hall.org`

We encourage you to read the tpc-faq and join with us in this grand experiment in providing local service on a global basis.

CIDR and the Evolution of the Internet Protocol

by

Hans-Werner Braun, San Diego Supercomputer Center

Peter S. Ford, Los Alamos National Laboratory

Yakov Rekhter, T. J. Watson Research Center, IBM Corp.

[Ed.: This article is reprinted from a paper presented at *INET 93*, the Internet Society conference, with permission.]

Abstract

The tremendous growth of the Internet has resulted in several efforts addressing the need to be able to address and route the global public Internet. Many of these efforts call for large changes in the deployed Internet infrastructure. We believe a prudent course of action includes extending the usable life of the *Internet Protocol Version 4* (IPv4) while evaluating the use of these new technologies. This requires incremental changes in the IPv4 routing technology base, but more importantly it will require additional coordination and management among the Internet service providers. We also believe that extending the life of IPv4 includes getting higher utilization of the IPv4 address space. Several strategies, mostly administrative in nature, are proposed to conserve the use of IP address space.

Introduction

The Internet is facing serious challenges with distinct (but interrelated) technical problems and several management problems which need to be addressed.

The three technical problems which need to be addressed are:

- Routing table expansion beyond router capabilities.
- Premature exhaustion of assigned IP network numbers due to inefficient allocation of the IPv4 address space.
- IP's inability to address more than 4 billion hosts connected to a single public internet.

Each of these technical problems has separate time horizons and each is subject to a variety of solutions.

The need for global operational coordination with a systems perspective is largely unfulfilled in the current Internet. Internet addressing and routing requires significant multilateral management to insure effective operation of the Internet infrastructure.

Current "save the Internet" proposals (PIP [6], SIP [4], TP/IX [14], TUBA [3, 9]) attempt to solve all of the above problems with the application and deployment of new, immature and certainly untested technology. Furthermore, some of these proposals fail to recognize that the problems of building a large global public internet cannot be solved strictly in the technical realm. Since the problem of routing table size may become serious in the near term (1–2 years), there is a perception that it is necessary to adopt one of the proposed new technologies as soon as possible.

We propose a two-pronged system approach to improve and enhance the current Internet. This approach combines technical and managerial aspects in a coherent fashion, addressing each of the above problems as they arise in an incremental manner. This greatly minimizes the risk of service disruption in the operational Internet and will allow time for much greater maturation of the proposed technologies and a better understanding of the requirements to be met.

CIDR as a solution to routing table size

The routing table size problem has several facets. A primary concern which led to the development of *Classless Inter-Domain Routing* (CIDR) [5, 7] is the imminent exhaustion of the class B address space. In the absence of CIDR, sites with more than 254 hosts need to advertise multiple class C networks into the Internet routing system. By using hierarchical routing, CIDR allows these sites to use a single prefix to advertise reachability of multiple class C networks, and subsequently permits full utilization of the class C address space with a minimum increase in routing table size.

CIDR can also be used recursively to hierarchically summarize routing information for multiple sites [10]. CIDR summarization can be done along provider based, geographically based or other hierarchical lines. Thus, routing table growth can scale in proportion to the number of elements within the chosen hierarchy (e.g., number of providers) instead of to the number of sites. To achieve such scaling with CIDR requires address assignment along hierarchical boundaries, which will require additional management of the address space.

Efficient use of address space

The class-oriented structure of IP addresses has led to fairly inefficient use of the IP address space. IP address space has been freely traded off for ease in routing administration. This tradeoff occurs in several ways:

- Many sites receive far more address space than they need. Many sites with much less than 64 thousand hosts use a single class B address in place of multiple Cs.
- Because of limited subnetting capabilities, many sites end up needing multiple network numbers (e.g., many universities have several class B networks where they use 8 bit subnet masks with very low occupancy on most subnets.)
- Many sites request additional network numbers simply to reduce the need to administratively coordinate within a campus.
- Many sites using IP are not accessible on the public Internet yet they use a significant portion of the current address space. (e.g., out of 44,000 assigned network numbers only 25 percent are reachable through the Internet).

Since it is now clear that IP has become a key component of the global Internet, and the IP address space is an increasingly scarce resource, it is necessary to judiciously allocate and assign the IP address space. The various addressing registries of the Internet must now incorporate guidance regarding Internet routing issues into their IP address assignment policies. As part of CIDR deployment, such efforts have already started at the U.S. NIC (Network Information Center) and the European RIPE NCC (Network Coordination Center), and CIDR addressing guidelines have been published [8].

Renumbering

The simplest hierarchy to use in a CIDR system is provider based addressing which allows hierarchical summarization of subscriber prefixes into a small number of routing entries. Provider based addressing is simple to administer since Internet subscribers will often get their addresses from a network service provider when they initially attach to the Internet.

When users switch providers, they will need to eventually change their addresses so that the routing system does not revert to "flat routing." Renumbering is often perceived as "impossible" to ask sites and Internet users to do.

continued on next page

Evolution of the Internet Protocol *(continued)*

However, going beyond the perception we observe that most Internet users and systems use names and the *Domain Name System* (DNS) [11] instead of addresses, limiting the exposure of host renumbering to system and network administrators. Internet users would certainly prefer name-based systems which either autoconfigure from a network service or at least are more easily configured than today's systems. There should be a coordinated effort to make administration of Internet systems easier and at the same time address renumbering as a solution to some of the scaling issues of the Internet within the context of IPv4. These activities could take place in the *Internet Engineering Task Force* (IETF) within the *Dynamic Host Configuration Protocol* (DHCP) and DNS working groups.

Renumbering of the current set of allocated class A, B and C network addresses will occur as necessary to increase the actual aggregation of network announcements. This will maximize the utility of the CIDR routing and addressing system. Renumbering will also recover significant amounts of unused IP address space, thus alleviating the problem of exhaustion of assigned IP network numbers. The combination of CIDR and renumbering makes it possible to use IPv4 until the total number of hosts approaches the practical limits of the 32 bit address field.

Network numbers for private internets

Hosts within sites that use IP can be partitioned into 3 three distinct categories:

- Hosts that do not require Internet access
- Hosts that need access to a limited set of Internet services (e.g., E-mail) which can handled by application layer relays.
- Hosts that need unlimited access to the Internet.

Only hosts in the last category require IP addresses that are globally unique. Hosts within the first and second categories may use IP addresses that are unique within a site, but may be globally non-unique. It is common for organizations to build private internets which have little or no hosts falling into the third category. Therefore, to conserve IP network address space utilization for the public Internet, we propose the allocation of specific IP network address blocks to be used by sites to identify hosts of type 1 or 2 within their private networks. These IP addresses will not be routed in the public Internet. Thus a site could assign two subnet addresses to each physical network. Systems either get a global address or a local-only address, depending upon what sort of access it needs. With the proposed scheme many large corporate sites (BBN, DEC, GE, IBM) can use a few class C network numbers from the global IP address space.

Management as a key component

Routing of the Internet used to be quite simple when the Internet was a single community of interest network administered by a single entity. As the Internet has grown to serve many other communities in a production mode of operation, the problems of administering the Internet routing and addressing has become more difficult.

As noted above, many of the problems in deploying CIDR and securing maximal use of IPv4 technology are related to administrative issues, such as hierarchical assignment of addresses, renumbering sites as necessary, and management of routing databases [1, 2] which contain route aggregation information. Multilateral coordination is just getting started in the Internet and the establishment of inter-provider operational procedures is underway.

The problems of multilateral coordination will be present for any of the proposed IP "next generation" proposals and we need to carefully observe and learn from CIDR deployment and operation to help facilitate future deployments of new technology.

Problems in large scale internetworking require a two-pronged systems oriented approach including both technical and management elements. Experience with the current Internet, specifically the inter-domain routing system engineering and management, shows that such an approach is necessary to deliver a workable solution. System-level interaction between technical and management components is required for a successful deployment of Internet technology. Currently, technical groups often know too little about actual operational issues, while the operational groups in turn often know too little about the feasibility of certain technologies. This state of affairs can be improved with better communication and interaction between these groups in a joint effort to reduce the risk of major operational problems in the Internet.

We observe that a key element to the success of any new strategy for the evolution of the Internet must address the issue of distributed management of the Internet system beyond simple addition of new technology to be deployed in the Internet. The Internet is beginning to move from centralized (e.g., NSFNET/Merit) towards distributed management (e.g., an open *Network Operations Forum* (NOF)), with an according redistribution of responsibility among multiple participants. A small number of entities controlling the future of the Internet is anachronistic and reminiscent of the concept of a single global telephone company.

CIDR rollout activities

The Internet community has already made significant plans for CIDR deployment laying the foundation for a successful roll out of BGP-4 when it becomes available during the summer of 1993. Here are some of the highlights:

- BGP-4 specification (IETF BGP working group)
- Vendor commitment to implement BGP-4: ANS, Cisco, Proteon, Wellfleet, 3Com, BBN. (All of these companies participate in BGP-4 deployment coordination meetings.)
- Coordination and planning for BGP-4 deployment
- U.S. Regionals workshop on CIDR
- Special meeting by "backbone" providers (INTEROP 93 Spring)
- IETF BGP deployment WG meetings
- Routing Database work: RIPE NCC and Merit.
- Routing Database Deployment: RIPE NCC, Merit, and CIX.
- Address assignment plans: U.S. Regionals and the U.S. NIC, RIPE NCC and EBONE, CIX, Japan, etc.
- Initial hierarchical allocation of IP addresses as blocks of class Cs is being done by the RIPE NCC, and the U.S. NIC in conjunction with the IP network providers in the U.S.

Conclusion

The Internet can evolve incrementally to address two of the major problems, routing table expansion and premature exhaustion of assigned IP network numbers.

continued on next page

Evolution of the Internet Protocol *(continued)*

Small technological changes in the routing technology are being implemented by router vendors making BGP-4 [12] available during Summer 1993 for deployment in the Internet inter-domain routing system. To effectively use BGP-4 will require additional administrative effort within the Internet. Most of this effort will be by Internet service providers, but the ability to reduce sites' efforts in changing addresses may require additional effort within the IETF.

CIDR and renumbering sites according to a CIDR management plan needs to be articulated to the Internet community as the most cost effective and risk adverse in contrast to installing new host and router technology throughout the Internet. The latter will be necessary only as the total number of hosts in the Internet approaches the limits of the 32 bit address field.

From the outset "saving the Internet" requires much more than a technical mechanism—it requires effective *management.* Deploying new or enhanced protocols to address the shortcomings of existing protocols is an important subset of the overall problem. However, such deployment can not occur in isolation and without proper development and deployment planning and management.

What began as an ARPANET research testbed has grown into a vast international fabric which must rely on stable, soundly-managed, underlying technology. Therefore, we need to focus on the stability of the operational environment of the existing and evolving Internet fabric. The current hype over the next generation of IP as an immediate requirement for a graceful infrastructural evolution is at best unnecessary, at worst dangerous, since it creates a poor public perception of the Internet and may lead to degraded Internet service. The next generation of IP is not really needed as urgently as some would like us to think. There is no reason to prematurely abandon the use of IPv4 until we can gain sufficient development and operational experience and develop consensus on a new network layer protocol, so as to address the problem of identifying more than 4 billion hosts in the public Internet.

The IAB/IESG/ISOC would err in making rash decisions regarding crucial changes in the Internet technology base without the development of adequate consensus from all affected communities (user, operator, developer). We must accept and cultivate the reality of the Internet as a cooperative enterprise and encourage coordination among entities participating in the growth and evolution of the Internet.

The authors would like to thank Ross Callon (Wellfleet) and Elise Gerich (Merit) for helpful conversations. Peter Ford acknowledges support from the U.S. Department of Energy's Office of Energy Research (Contract No. KC0702) and Los Alamos National Laboratory (Contract No. W-7405-ENG-36). The authors also thank the National Science Foundation for continued support for working on the NSFNET project. The work of Yakov Rekhter was supported by the National Science Foundation, under contract NCR-92-19216. Views and conclusions expressed in this article are not necessarily those of the National Science Foundation.

References

[1] Adams, A., "The Merit Policy-Routing Configuration System," *ConneXions,* Volume 7, No. 2, February 1993.

[2] Bates, T., Jouanigot, J., Karrenberg, D., Lothberg, P., Terpstra, M., "Representation of IP Routing Policies in the RIPE Database," File `ripe-81` available via FTP `from ftp.ripe.net`.

[3] Callon, R., "TCP and UDP with Bigger Addresses (TUBA), A Simple Proposal for Internet Addressing and Routing," RFC 1347, June 1992.

[4] Deering, S., "SIP.: Simple Internet Protocol," *IEEE Network,* May 1993.

[5] Ford, P., Braun, H.-W., Rekhter, Y., "Improving the Routing and Addressing of IP," *IEEE Network,* May 1993.

[6] Francis, P. "A Near-Term Architecture for Deploying Pip," *IEEE Network,* May 93.

[7] Fuller, V., Li, T., Yu, J., Varadhan, K., "Supernetting: an Address Assignment and Aggregation Strategy," RFC 1338, June 1992.

[8] Gerich, E. P., "Guidelines for Management of IP Address Space," RFC 1366, October 1992.

[9] Katz, D., Ford, P., "TUBA: Replacing IP with CLNP," *IEEE Network,* May 1993.

[10] Kleinrock, L., Farouk, K., "Hierarchical Routing for Large Networks," *Computer Networks,* North-Holland Publishing Company.

[11] Mockapetris, P. V., "Domain names—implementation and specification," RFC 1035, November 1987.

[12] ed. Rekhter, Y., Li, T. "A Border Gateway Protocol 4 (BGP-4)," Internet Draft, December 1992.

[13] Rekhter, Y., Li, T. "An Architecture for IP Address Allocation with CIDR," Internet Draft, February 1993.

[14] Ullman, R. L., "TCP/IP: Internet Version 7," RFC 1475, June 1993.

[15] Crocker, D., "The ROAD to a New IP," *ConneXions,* Volume 6, No. 11, November 1992.

Learn more in Session S19: "The Future of IP" Wednesday at 3:30pm.

HANS-WERNER BRAUN joined the San Diego Supercomputer Center (SDSC) as a Principal Scientist in January 1991, focusing on NSF funded efforts for NREN engineering and network performance related research. His prior work included being a Principal Investigator on the NSFNET backbone project while working at Merit, Inc., in Michigan. After graduating in 1978 in West Germany, holding a Diploma in Engineering he worked for several years at the Regional Computing Center of the University of Cologne in West Germany on network engineering. He participates in several national and international networking related committees. His e-mail address is: `hwb@sdsc.edu`

YAKOV REKHTER holds M.S. in Physics from St. Petersburg University, Russia, M.S. in Computer Science from New York University, and Ph.D. in Computer Science from Polytechnic University. He has been with IBM since 1984, and is currently a Research Staff Member and a manager of High-Performance Communication group at T. J. Watson Research Center. Dr. Rekhter was one of the leading architects, as well as the major software developer, of the NSFNET Backbone Phase II. Present activities include work on Classless Inter-Domain Routing (CIDR), work on the Unified Approach to Inter-Domain Routing (under the contract with the National Science Foundation), work on supporting host mobility, and work on supporting IP over Fibre Channel subnetworks. His e-mail address is: `yakov@watson.ibm.com`

PETER SEWALL FORD is a member of the technical staff at Los Alamos National Laboratory where he is currently working with the U.S. National Science Foundation on the NSFNET and developing software to manage and instrument an 800 Mbit/sec crossbar network. Mr. Ford has also worked at Los Alamos' Center for Nonlinear Studies, on UNIX systems for the Computer Science Department at the University of Utah, and on distributed database systems and languages at Britton-Lee, Inc. His university degree is in General Studies from the University of Michigan. E-mail: `peter@goshawk.lanl.gov`

A User's View of the Next Generation of IP (IPng)

by Eric Fleischman, Boeing Computer Services

Introduction

The activities within the Internet community to resolve the scaling problems of the Internet Protocol (IP) have been well documented by this and other publications (e.g., *ConneXions*, November 1992, Volume 6, Number 11 "The ROAD to a new IP" by David Crocker). These articles have detailed the specific proposals which are being examined as potential replacements for the current version of IP, IP version 4 (IPv4), and its routing protocols. This quest to identify the future version of IP, together with the list of contending replacement protocol solutions, is often generically termed the *Internet Protocol Next Generation* and abbreviated "IPng." [Note: The term "IPng" has recently replaced the older term "IPv7" to become the proper reference for the potential IPv4 replacement protocols. The coining of the new term was motivated by the fact that the TP/IX IPng proposal has now been assigned the IP version number of seven. Thus, TP/IX is now IPv7.]

Viewpoints

However, no article to date has examined the implications of IPng to the current TCP/IP user community. The goal of this article is to present a user's viewpoint on the IPng issues. This "user's view of IPng" must be presented with a large dose of humility. Humility is necessary because the various issues which pertain to the selection of a scalable IP replacement protocol are quite complex. These issues represent a multi-faceted problem domain which may be profitably examined from many pertinent viewpoints. Among these viewpoints are the following:

- Network Service Providers
- Computer/Network Research Community
- Vendors of Computer/Network Products. This community may be further subdivided into vendors of "Intermediate Systems" products and vendors of "End Systems" products.
- End Users (i.e., entities who use computers as an "overhead expense" to accomplish a non-computer-oriented goal). Several different end-user perspectives may be distinguished including private citizens, government, education, and industry.

Each of these diverse viewpoints provide valuable perspectives for evaluating the IPng problem domain. Each perspective potentially asks different questions which are highly relevant for the ultimate identification of the "best" IPng proposal. However, the sheer number of these differing viewpoints complicates the IPng evaluation process by adding additional concerns and requirements. This process is further complicated by the fact that different entities sharing the same "viewpoint" are themselves subject to different priorities due to "corporate culture" distinctives. For example, one would expect a small corporation to have a different list of priorities than a very large corporation even though both viewpoints are "industry" viewpoints. Similarly, entities perpetually on the technological "bleeding edge" will naturally view things differently than technologically conservative entities.

Because of the complexity of the larger context, this article will examine the implications of IPng from the point of view of a single Fortune 100 corporation which has heavily invested in TCP/IP technology in order to achieve its business goals. While this is merely a view from a single corporation, it is hoped that this viewpoint may prove to be generally relevant to the Internet Community as a whole.

Characteristics

The following five key characteristics describe our environment and are probably representative of other large TCP/IP deployments. We believe that understanding these characteristics is very important for obtaining insight into the implication of IPng to large user populations.

• *Host Ratio:* Many corporations explicitly try to limit the number of their TCP/IP hosts that are directly accessible from the Internet. This is done for a variety of reasons (e.g., security). While the ratio of those hosts that have direct Internet access capabilities to those hosts without such capabilities will vary from company to company, ratios ranging from 1:1000 to 1:10,000 (or more) are not uncommon. In addition, not all corporations with TCP/IP deployments currently have Internet attachments. The implication of this point is that the state of the world-wide (IPv4) Internet address space only directly impacts a tiny percentage of the currently deployed TCP/IP hosts within a large corporation. This is true even if the entire population is currently using Internet-assigned addresses.

• *Router-to-Host Ratio:* Most corporations have significantly more TCP/IP hosts than they have IP routers. Ratios ranging between 100:1 to 600:1 (or more) are common. The implication of this point is that a transition approach which solely demands changes to routers is generally much less disruptive than an approach which demands changes to both routers and hosts.

• *Business Factor:* Large corporations exist to fulfill some business purpose such as the construction of airplanes, baseball bats, cars, or some other product offering. Computing is merely a tool to help automate business processes in order to more efficiently accomplish the business goals of the corporation. Automation is accomplished via applications. Data communications, operating systems, and computer hardware are simply the tools used by applications to accomplish their goals. Thus, users actually buy applications and not networking technologies. The central lesson of this point is that IPng will be deployed according to the applications which use it and not because it is a better technology.

• *Integration Factor:* Large corporations currently support many diverse computing environments. This diversity limits the effectiveness of a corporation's computing assets by hindering data sharing, application interoperability, "application portability," and software reusability. The net effect is stunted application life cycles and increased support costs. Data communications is but one of the domains which contribute towards this diversity. For example, The Boeing Company currently has deployed at least sixteen different protocol families within its networks (e.g., TCP/IP, SNA, DECnet, OSI, IPX/SPX, AppleTalk, XNS, etc.). Each distinct Protocol Family population potentially implies unique training, administrative, support, and infrastructure requirements. Consequently, corporate goals often exist to eliminate or merge diverse Data Communications Protocol Family deployments in order to reduce network support costs and to increase the number of devices which can communicate together (i.e., foster interoperability). This results in a basic abhorrence to the possibility of introducing "Yet Another Protocol" (YAP). Consequently, an IPng solution which introduces an entirely new set of protocols will be negatively viewed simply because its by-products are more roadblocks to interoperability coupled with more work, expense, and risk to support our computing resources and business goals.

continued on next page

A User's View of IPng *(continued)*

Having said this, it should be observed that this abhorrence may be partially overcome by "extenuating circumstances" such as applications using IPng which meet critical end-user requirements or by broad (international) commercial support.

• *Inertia Factor:* There is a natural tendency to continue to use the current IP protocol (IPv4) regardless of the state of the Internet's IPv4 address space. Motivations supporting inertia include the following: existing application dependencies (including *Application Programming Interface* (API) dependencies); opposition to additional protocol complexity; budgetary constraints limiting additional hardware/software expenses; additional address management and naming services costs; transition costs; support costs; training costs; etc. As the number of our deployed TCP/IP hosts continues to grow towards the 100,000 mark, the inertial power of this population becomes increasingly strong.

However, inertia even exists with smaller populations simply because the cost to convert or upgrade the systems are not warranted. Consequently, pockets of older "legacy system" technologies often exist in specific environments (e.g., we still have pockets of the archaic BSC protocol). The significance of this point is that unless there are significant business benefits to justify an IPng deployment, economics will oppose such a deployment. Thus, even if the forthcoming IPng protocol proves to be "the ultimate and perfect protocol," it is unrealistic to imagine that the entire IPv4 population will ever transition to IPng. This means that should we deploy IPng within our network, there will be an ongoing requirement for our internal IPng deployment to be able to communicate with our internal IPv4 community. This requirement is unlikely to go away with time.

Costs and benefits

Thus, the central, bottom-line question concerning IPng from the user perspective is: What are the benefits which will justify the expense of deploying IPng? At this time we can conceive of only four possible causes which may motivate us to deploy IPng:

Possible Cause:	*Possible Corporate Response:*
• Many Remote (external) Peers solely use IPng.	Gateway external systems only.
• Internet requires IPng usage.	Gateway external systems only.
• "Must have" products are tightly coupled with IPng (e.g., "flows" for real-time applications).	Upgrade internal corporate network to support IPng where that functionality is needed.
• Senior management directs IPng usage.	Respond appropriately.

It should explicitly be noted that the reasons which are compelling the Internet Community to create IPng (i.e., the scalability of IPv4 over the Internet) are not themselves adequate motivations for users to deploy IPng within their own private networks. That is, should IPng usage become mandated as a prerequisite for Internet usage, a probable response to this mandate would be to convert our few hosts with direct external access capabilities to become IPng-IPv4 application-layer gateways (i.e., dual stacks). This would leave the remainder of our vast internal TCP/IP deployment unchanged. Consequently, given gateways for external access, there may be little motivation for a company's internal network to support IPng.

Motivations

We suspect that there are only two causes which will motivate users to widely deploy IPng:

(1) If IPng products add critical functionality which IPv4 can't provide (e.g., real time applications, multimedia applications, genuine (scalable) plug-and-play networking, etc.), users would be motivated to deploy IPng where that functionality is needed. However, these deployments must combat the "Integration Factor" and the "Inertia Factor" forces which have previously been described. This implies that there must be a significant business gain to justify such a deployment. While it is impossible to predict exactly how this conflict would "play out," it is reasonable to assume that IPng would probably be deployed according to an "as needed only" policy. Optimally, specific steps would be taken to protect the remainder of the network from the impact of these localized changes. Of course, should IPng become bundled with "killer applications" (i.e., applications which are extremely important to significantly many key business processes) then all bets are off: IPng will become widely deployed.

(2) Should IPng foster a convergence between Internet Standards and International Standards (i.e., OSI), this convergence could change IPng's destiny. That is, the networks of many large corporations are currently being driven by sets of strong, but contradictory, requirements: one set demanding compliance with Internet Standards and another set demanding compliance with International Standards. [Note: The following is a single example concerning why International Standards are important to large corporations. Corporations conducting a global business are subject to the regulations of those countries in which they trade. International commerce is regulated by governments, many of whom have placed restrictions upon data communications. These restrictions affect the data communications of a corporation's products as well as the data communications between corporations (i.e., business partners, customers, and suppliers). International Standards are the only certain bet to comply with world-wide commerce restrictions.]

If a means could be found to achieve greater synergy (integration/adoption) between Internet Standards and International Standards then corporate management may very well be inclined to mandate internal deployment of the merged standards and promote their external use. Optimally, such a synergy should offer the promise of reducing currently deployed protocol diversity (i.e., supports the "Integration Factor" force). Depending on the specific method by which this convergence is achieved, it may also partially offset the previously mentioned "Inertia Factor" force, especially if IPng proves to be a protocol which has already been deployed.

Consequently, mandating IPng to communicate over the Internet does not correspondingly imply the need for large corporations to generally support IPng within their networks. Thus, while the IPv4 scalability limitations are compelling reasons to identify a specific IPv4 replacement protocol for the Internet, other factors are at work within private corporate networks. These factors imply that large TCP/IP end users will have a continuing need to purchase IPv4 products even after IPng products have become generally available.

User requirements

This article began with an acknowledgment that the Internet community is composed of a variety of members who possess potentially differing viewpoints of the IPng problem domain.

continued on next page

A User's View of IPng *(continued)*

We perceive that our vantage point would identify the following as critical end-user requirements for IPng:

- The IPng approach must permit users to slowly transition to IPng in a piecemeal fashion. Even if IPng becomes widely deployed, it is unrealistic to expect that users will ever transition all of the extensive IPv4 installed base to IPng. Consequently, the approach must indefinitely support corporate-internal communication between IPng hosts and IPv4 hosts regardless of the requirements of the world-wide Internet.
- The IPng approach must not hinder technological advances to be implemented (e.g., mobile hosts, multimedia applications, or real-time applications).
- The IPng approach is expected to eventually foster greater synergy (integration/adoption) between Internet Standards and International Standards (i.e., OSI). [Note: This may be accomplished in a variety of ways including having the Internet Standards adopted as International Standards or else having the International Standards adopted as Internet Standards.]
- The IPng approach should have "self-defining network" (i.e., "plug and play") capabilities. That is, large installations require device portability in which one may readily move devices within one's corporate network and have them autoconfigure, autoaddress, autoregister, etc. without explicit human administrative overhead at the new location—assuming that the security criteria of the new location have been met.

Conclusion

In summary, the key factor which will determine whether—and to what extent—IPng will be deployed by large end users is whether IPng will become an essential element for the construction of applications which are critically needed by our businesses. If IPng is bundled with applications which satisfy critical business needs, it will be deployed. If it isn't, it is of little relevance to the large end user. Regardless of what happens to IPng, the large mass of IPv4 devices will ensure that IPv4 will remain an important protocol for the foreseeable future and that continued development of IPv4 products is advisable.

References

[1] Crocker, D., "The ROAD to a New IP," *ConneXions,* Volume 6, No. 11, November 1992.

[2] Lottor, M., "Internet Growth (1981–1991)," RFC 1296, Jan. 1992.

[3] Solensky, F., "The Growing Internet," *ConneXions,* Volume 6, No. 5, May 1992.

[4] desJardins, R., "Internet 2000," *ConneXions,* Volume 6, No. 10, October 1992.

Learn more in Session S19: "The Future of IP" Wednesday at 3:30pm.

ERIC FLEISCHMAN is a Senior Principal Scientist (Network Architect) within Boeing Computer Services' Delivery Systems Architecture and Technical Planning group. He is currently participating in a number of corporate-wide technology assessment and system integration activities. Previous to this he was actively involved in designing the Boeing Enterprise Network and assisted several of Boeing's strategic computer vendors in the design of their network products and technologies. Eric has previously worked as a Software Engineer for Victor Technologies (Scotts Valley, Ca.) and Digital Research (Monterey, Ca.) and was a Member of the Technical Staff at AT&T Bell Laboratories (Columbus, Ohio). He has degrees from Wheaton College (Illinois), University of California at Santa Cruz, and the University of Texas at Arlington. He is a member of the IEEE and ACM and is a participant in several IETF working groups. He may be reached via e-mail as: `ericf@atc.boeing.com`

The Internet as a tool for Business Communications?

by John Curran, NEARnet/BBN Systems and Technologies

Introduction

In a handful of years, the Internet has evolved from an experimental network serving select academic and research institutions into a richly connected network spanning the world. With its vast connectivity and proven interoperability, the Internet is being recognized as an invaluable tool for business communications. This article presents a summary of the Internets unique advantages for business communication, and highlights ongoing work in performance, security, and billing services needed for continued commercial growth.

The value of the Internet

Businesses today use a variety of communications tools (such as telephone, fax, and courier services) to reach their clients and suppliers. Given the popularity of these methods, why should a business connect to the Internet? What differentiates Internet communications from other communications? These are the questions that must be addressed before an organization will invest in establishing an Internet connection. In order to understand the value of Internet service, two essential aspects of the Internet must be explored: connectivity and interoperability.

Connectivity

Through the interconnection of over twelve thousand computer networks (and over 1.7 million computer systems [1]), the Internet provides connectivity that is unmatched by any private network service. Internet services allow for seamless data communications to thousands of suppliers and customers without having to establish *in advance* the exact organizations that may be reached. In this manner, the Internet resembles the international phone system where by a call may be placed to almost any destination without prior arrangements being required. This ability to perform ad hoc data communications is a feature unique to the Internet.

As an example, there has recently been a trend towards using *Bulletin Board Systems* (BBS) to achieve data communications with customers [2]. Many businesses are distributing product literature, software and problem fixes via such systems. While these systems are effective at tapping into present base of modem-equipped PCs, having to manage the resulting telephone lines, modem pool, and user IDs can be quite burdensome. A corporate BBS for communication with customers is a *closed* system, in that only those customers that have been supplied with the appropriate phone numbers and user IDs can make use of the system. In the Internet, standard paradigms for distributing information make it easy for present customers (and more importantly, millions of potential customers) to access designated information without any prior arrangements having been made.

Interoperability

While the Internet's connectivity is impressive, the true value of the Internet is the ability to easily *interoperate* with any (and every) other Internet system. This high level of interoperability is achieved by using the TCP/IP protocol suite for communication. Unlike many other protocol suites, TCP/IP was designed to support heterogeneous networks and has been refined over years of multivendor operations. [3] As a result, TCP/IP has become a de facto standard for integrating diverse networks. TCP automatically adjust the transmission rate to obtain the ideal performance based on the overall end-to-end conditions. This ability to span diverse networks is essential in the Internet today, where individual links in the network vary from 1200 bps to 45 Mb/s.

continued on next page

Internet as a tool for Business *(continued)*

This high degree of interoperability is not restricted to the network layer of the Internet. Through the same open standards process that created TCP/IP, the *Internet Engineering Task Force* has standardized many of the application protocols in use today. Some of the more popular protocols include interactive login (Telnet), file transfer protocol (FTP), and mail (SMTP). Due to extensive development and testing activities *before* standardization, interoperability is readily achieved between different implementations of these protocols. TCP/IP applications enjoy an unequaled reputation for interoperability and are often chosen for use in private networks on this basis alone.

The needs of a business customer

The tremendous commercial interest in the Internet has led network service providers to further explore the needs of the business customer. Many businesses have expectations in the areas of security, performance, and billing which are not directly addressed by the current Internet service offerings.

Internet security

One of the first issues that organizations face when connecting to the Internet is: "Is the Internet secure?" While that question may seem simple, the demand for (and often the definition of) security varies between organizations. For this reason, it is important for businesses to compare the security features of the Internet against their particular needs before making use of the Internet for their business communication. In particular, the business' need for authentication and confidentiality must be carefully considered.

Authentication is the ability to identify the requester of a particular service. Once the requester of a service is known, it is possible to validate against a list of permitted users for a given network service. Clearly authentication is a desirable function of a network service, and many sites which connect to the Internet have an expectation that authentication is an inherent feature of the network. In fact, the Internet (as a network layer service) does not provide any authentication of service requests. A given request (whether to access a file or to interactively connect to a computer) appears to the network layer as simply another datagram for delivery, and doesn't contain any explicit identification of the user or program requesting the service. As a result, it is common for applications to add their own validation process (for example, prompting for a username and password) where feasible, but the lack of a common authentication system results in inconsistent approaches to validation, and makes it quite difficult to track suspect usage. [4]

The Internet is much like the Public Switched Telephone Network: when a phone call arrives, it is not identified as a given person. At best, the call is identified by its originating phone number. While there is often a relationship between the originating phone number and the caller, this is not always true (as is the case with public phones). Similarly, there are many systems in the Internet operated in an open manner, and hence a service request from the system does not imply anything about the requester. For this reason, it is not possible to rely on the originating address of the request for authentication, and businesses that require authentication must be prepared to implement it themselves.

Another major security concern for businesses is the *confidentiality* of their Internet connection. Most business activities require handling sensitive material of one form or another; it can be client lists, product development plans, or even something as routine as charge card information.

The inadvertent disclosure of such information can ruin business relationships and/or result in significant corporate liability. While the majority of network services which make up the Internet are operated in a secure manner, that is not a guarantee of confidentiality. Businesses using the Internet today must balance their need to communicate against the potential risk of disclosure, and this limits the applicability of the Internet at present to a subset of possible business uses.

There are several initiatives underway to improve security in the Internet. For electronic mail, the *Privacy Enhanced Mail* (PEM) standards will allow for confidential and authenticated mail. Electronic mail sent via PEM can be "signed" by the author using a personal key and then verified by the recipient of the message using a public key. [5] Commercial products based on PEM are under development now, as are the necessary services to distribute public/private key pairs to users. As this PEM infrastructure develops, companies will be able to use PEM to conduct business over the Internet in an authenticated and confidential manner.

While PEM will protect mail (and mail-based applications), there is another activity underway which will protect non-mail applications. The *Trusted Network Computing Environment,* recently announced by Novell, promotes a distributed mechanism for encrypting local and wide-area communication [6]. This computing environment uses public-key technology to distribute DES session keys in a secure manner. While network-layer encryption products are already available, the early adoption of a scalable, common architecture will speed the deployment of these systems in the Internet, and enable transmission of business critical data over the Internet.

Internet performance

Organizations purchasing Internet services often have performance expectations in the areas of throughput and reliability. Depending on the business application, these expectations can be fairly flexibility or quite demanding. Electronic mail, for example, is a store-and-forward, low-bandwidth protocol which makes few demands on the underlying network. On the other hand, file sharing across the Internet requires both high throughput and high reliability if it is to be successful. Businesses which do not take the Internet's throughput and reliability capabilities into consideration run the risk of unsuccessful communications and customer dissatisfaction.

For example, it is not possible to guarantee the rate of data transfer (or *throughput*) across the Internet. The reason for this is simple: the Internet serves thousands of organizations using shared resources, and lacks any mechanism for reserving bandwidth. If bandwidth is scarce due to sudden increase in demand, each active user will experience a drop in performance. A performance drop will hardly be noticed by the typical electronic mail and news participant, but can significantly delay file transfers and remote login applications. As a result, using the Internet for time critical applications requires careful research to determine the extent of shared resources along the communications path.

Another important consideration in business communications is *reliability*. Reliability means predictable performance over time, and in the Internet this can be measured by both the overall availability of the network and the availability of the support services which make the Internet useful.

continued on next page

Internet as a tool for Business *(continued)*

The Internet is not a single network being operated by a single organization. As a large collection of interconnected (but distinct) TCP/IP networks, coordination between Internet service providers is essential for assuring end-to-end availability of service. While each provider can control the components of their network, it is only by working together that overall reliability can be achieved. Problem resolution across network boundaries requires coordination, and is often slower than problem resolution within a given network. Businesses considering the Internet for critical applications should consider both the number and variety of the network providers involved before deciding, and may want to arrange for a single entity to accept operational responsibility.

There are a number of efforts in progress to improve communications between network providers, and day- to-day operational coordination has increased as a result. Significant efforts include the *User Connectivity Problems* and *Network Joint Management* working groups of the IETF. These groups are researching inter-provider trouble ticket systems and inter-provider operational communications. [7] One tangible benefit of this work is that Internet sites can report a problem that spans multiple providers with increasing confidence that it will be transferred to the appropriate provider for resolution.

Support for Internet Commerce

The Internet is increasingly being used for product delivery, where the product is *information*. Electronic news, books, and technical information are available via the Internet from multiple sources under a variety of billing arrangements. As more information services use the Internet to distribute their products, the demand for network-based ordering, billing, and payment services will increase dramatically.

Some communications networks already provide billing services to information providers. For example, many telephone carriers provide content billing for "900" numbers. Despite the wealth of information available on the Internet, no comparable method of billing exists. Each information provider either does not charge (the most common case) or pre-authorizes particular systems for access. This lack of network-based billing services has resulted in an abundance of "free" information on the Internet, but much of this information is out of date or poorly organized since there is no cost recovery for maintenance.

Businesses considering providing services over the network should be prepared to handle their own ordering and billing requirements for the time being, although there is significant work underway to provide to such services. One of the more mature efforts, MCCs *Enterprise Integration Network* (EInet), is now providing value-added Internet services including secure product directories and product ordering, with on-line payment anticipated in the future. [8] Another indication of interest in this area is the recent IETF Mercantile Protocols BOF meeting to discuss protocol requirements for on-line commerce.

Conclusion

Before choosing the Internet as a business communications tool, it is important to understand both the Internet's capabilities in these key areas and an organization's needs. Internet services are being used to gain strategic advantage over competitors in areas such as customer support, product delivery, vendor communications, and improving staff technology awareness.

Businesses must remain aware of technical and service developments in the Internet industry, or risk missing opportunities. Finally, incorporating the emerging needs of businesses into the future direction of the Internet is crucial to the long term usefulness and viability of the Internet as a tool for business communications.

References

[1] Lottor, M., "Internet Domain Survey," July 1993.

[2] Harrer, J., "Bulletin Board Systems are Beginning to Emerge in Business Applications," *Telecommunications,* March 1993.

[3] DDN Protocol Handbook, Volume 2, "The DARPA Internet Protocol Suite," pg. 2–27.

[4] Holbrook, P., Reynolds, J., "Site Security Handbook," RFC 1244.

[5] Galvin, J., "The Deployment of Privacy Enhanced Mail," *ConneXions,* Volume 5, No. 10, October 1991.

[6] *Network World,* July 19 1993, "Novell to head LAN security movement," pg. 7.

[7] Johnson, D., "Network Operation Center Trouble Ticket Requirements," RFC 1297.

[8] *Communications Week,* June 21 1993, "Sprint & MCC team on Electronic Commerce," pg. 5.

[9] Crocker, D., "Evolving the System," in *Internet System Handbook,* edited by Daniel Lynch and Marshall Rose, Addison-Wesley, Reading, Massachusetts, 1993, ISBN 0-201-56741-5.

[10] Chapin, A. L., "The Internet Standards Process," RFC 1310, March 1992.

[11] Lottor, M., "Internet Growth (1981–1991)," RFC 1296, January, 1992.

[12] Solensky, F., "The Growing Internet," *ConneXions,* Volume 6, No. 5, May 1992, pp 46–48.

Learn more in Session S4: "Commercial Use of the Internet," Wednesday morning at 10:30am.

JOHN CURRAN is a Network Analyst in the Systems and Technologies Division of Bolt Beranek and Newman. He has worked on a variety of network projects including CSNET and the NSF Network Service Center (get NNSC). He is an active member of the Internet Engineering Task Force in the user services, operational requirements, and Internet areas. His current responsibilities include network analysis and product definition for the NEARnet suite of Internet services. E-mail: `jcurran@nic.near.net`

Write to *ConneXions*!

Have a question about your subscription? Are you moving, and need to give us your new address? Suggestions for topics? Want to write an article? A letter to the Editor? Have a question for an author? Need a *ConneXions* binder? Want to enquire about back issues? (there are now over 75 to choose from; ask for our free 1987–1992 index booklet and the 1993 partial index sheet). We want to hear from you. Send your questions, comments or suggestions to:

ConneXions—The Interoperability Report
480 San Antonio Road
Mountain View, CA 94040–1219
USA
Phone: +1 415-941-3399 or 1-800-INTEROP (Toll-free in the USA)
Fax: +1 415-949-1779
E-mail: `connexions@interop.com`

Issues in Internet Security

by Jeffrey I. Schiller, MIT

Introduction

Since the birth of the Internet, computer security concerns have taken a back seat to more "interesting" topics such as routing protocol design and higher speed networking. But as the network is used more and more to provide production services, the security of the information it carries is of more importance. Yet today, we still fail to see an integrated effective approach to data security widely deployed.

Today's practice

When people think of secure computing systems, they think about login screens and file access control lists. In essence these form the user interface seen by most people within a "secure" computing context.

"Secure" distributed computing schemes appear the same. Before you can use the system you must login. If you don't have the proper access rights to files, you cannot access them. Yet simply because a system has the user interface of a secure system, it doesn't mean that it is really secure.

In the more traditional timesharing model of computing. The human interface to the system was through a "dumb" terminal. All of the security relevant parts of the system were under the management and control of the system operators. Only if the system had a fundamental security flaw could it be compromised.

The introduction of *client–server computing* has changed all that. Now instead of a dumb terminal, a user accesses the system via a workstation or personal computer, a computer usually under their direct control and not that of the "Server" operator.

With such client–server systems, all security relevant functions *must* be embodied within the server systems and not within the client workstations. To understand why this is so, we need to look at some of the features of most network technologies.

Most local area networks are based on broadcast technology. When one system on the network sends data to another system, all the systems on the local area network cable receive the data, but the non-involved systems discard it from their network interface because the data was not addressed to them. Of course who is to say that an intruder controlled system will be so polite!

Another important feature of most local area technology is that it is impossible to *know* that a given piece of data originated from a particular system. Although in most systems every received piece of data is labeled with the network cable address of the originating system, this labeling cannot be trusted. It is up to the sending system to insert the proper address. There is no way for the receiving system to verify that the sending system is in deed who it claims to be.

What all this leads to is that a network server system cannot know by simple inspection of incoming network requests that those requests originated from only a given set of client systems. In other words a network request can as easily come from an intruder controlled system as from a legitimate system. Therefore if security is enforced within the client systems, it won't be enforced on the intruder's system!

A mistaken way this is often addressed is to require a login process for each client. Before a server will believe data packets originating from a particular client, a username and password need to be exchanged between client and server. This potentially suffers from two security problems. The first is simply that the password exchange can be viewed by any connected system (remember, they all receive all data at the lowest hardware level). The second is that once a user is logged in from a particular workstation, an intruder can send bogus network requests and label them as coming from that legitimate user's workstation. This type of attack has in fact been successfully used against a popular network operating system.

How to do it Right

The key to doing client–server computing securely is to first ensure that *authorization* decisions are made by the server software and that all client requests are properly authenticated.

The use of *cryptography* is crucial to authenticating client requests. Given that any client computer system can masquerade as any other, the only way to ensure that an intruder isn't inserting bogus network requests into the system is to make sure that to do so the intruder would first need a piece of information that is simply not available, the correct cryptographic key.

To securely authenticate information, each client system needs to have a secret cryptographic key (shared with the server) which is used to either completely encrypt all network requests and responses (if confidentiality is required) or to compute a cryptographic checksum of the network data exchange if only integrity needs to be ensured.

However it is easier to state that all clients should have such keys installed (securely of course!) then it is to arrange for this to in fact be the case. There are two basic techniques. The first involves the installation of the necessary cryptographic keys on the workstations themselves (probably via a manual process involving system managers installing manual keys). The second is to use an authentication system which distributes keys as part of the login process that each user goes through.

The idea here is to use the end user's password as a temporary secret key that is used to encrypt a "session" key which is downloaded at login time. A side effect of this technique is that if each logged in user is assigned a different session key, then any data checksummed using that key is known to have originated from the workstation of the logged in user.

Kerberos: Athena's answer

MIT's *Project Athena* addressed this problem in the design of the *Kerberos* authentication system. When a user of the Athena environment logs in, their password is used to encrypt a session key which is then used in later network transactions to authenticate and protect network requests.

To date MIT uses Kerberos to protect remote login access, remote file access and most recently the dissemination of student grades over the network (an encrypted application of course!).

The Open Software Foundation's *Distributed Computing Environment* (DCE) is using Kerberos as the underlying system for its secure remote procedure call and file system environment.

Public Key approaches

Public Key Cryptographic Systems offer a unique feature. As opposed to conventional encryption systems which use only one secret encryption key, Public Key Systems use two.

continued on next page

Issues in Internet Security *(continued)*

One key, called the *private key,* is maintained confidentially. The other key, called the *Public Key,* is not confidential. In fact it can be published widely. Information enciphered with the public key can only be deciphered with the private key, yet having knowledge of the public key does not yield sufficient knowledge to an intruder to determine the private key.

Public Key technology can be used by network authentication systems such as Kerberos. Doing so both improves the security and scalability of such systems. In fact MIT is looking into modifying a later version of Kerberos to take advantage of the RSA encryption system.

Digital Signatures

One of the more interesting uses of Public Key systems such as RSA is in the creation of *Digital Signatures.* A Digital Signature is a cryptographically generated bit sequence that is uniquely determined by the contents of a document and a private key. However once created, it can be verified by using a published public key. Such a signature permits the electronic equivalent of an ink signature for authenticating a document.

For example suppose I have a public/private key pair of which I make the public key widely known (and everyone knows that the public key in fact belongs to me). I can write an electronic document (say a bank check!) and create a Digital Signature of that document by using my private key. Anyone with my public key can then compare the document (the check) with the Digital Signature and know that only I could have created the document. For if the document itself is modified in any way, or the Digital Signature itself is tampered with, then the cryptographic operation to verify the signature (which only requires my Public Key, so can be performed by anyone) will fail.

Digital Signatures are one of the more important up and coming electronic security techniques. They will permit the widespread use of data networking for the transaction of normal business, potentially eliminating most documents that today must be printed and signed for authentication.

Governments: Enemies of Strong Security?

The key to client–server computing security is cryptography. But this technology has been slow to catch on, in part because of barriers erected by governments. Simply put cryptography enables several security services, including confidentiality. Yet governments often do not want their citizenry to have access to confidentiality so strong that the government cannot penetrate it when it wishes.

To slow down the proliferation of cryptographic devices, many governments heavily regulate it. The regulation may range from outright prohibitions to controls on exports of cryptographically based systems.

To deal with the regulations often involves expensive interactions with bureaucracies. Combine this hassle with the lack of perceived demand for strong security (non-strong security is often perceived to be sufficient until *after* a given organization has been attacked!), and it is easy to see why so little attention has so far been paid to security within commercial product offerings.

Of course as we move to a more electronically connected world, the need to use cryptographic security systems will force the review of this issue by many governments.

References

[1] *ConneXions,* Volume 4, No. 8, August 1990, "Special Issue on Network Management and Network Security."

[2] Schiller, J., "Kerberos: Network Authentication, *ConneXions,* Volume 4, No. 1, January 1990.

[3] W. Diffie, "The first ten years of public-key cryptography," *Proceedings of the IEEE,* 76(5):560–577, May 1988.

[4] W. Diffie & M.E. Hellman, "Privacy and authentication: An introduction to cryptography," *Proceedings of the IEEE,* 67(3):397–427, March 1979.

[5] R. L. Rivest, A. Shamir, & L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM,* 21(2):120–126, February 1978.

[6] W. Diffie & M.E. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory,* IT-22:644–654, 1976.

[7] Ronald L. Rivest, "Cryptography," In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science,* Volume 1, pages 719–755, Elsevier Science, 1990.

[8] Kaliski, B., "An Overview of Public-Key Cryptography Standards," *ConneXions,* Volume 6, No. 5, May 1992.

[9] National Bureau of Standards, "Data Encryption Standard," (FIPS Publication 46–1), January 1988.

[10] Millikin, M., "Distributed Computing Environments: Laying the foundation for the future of interoperability," *ConneXions,* Volume 4, No. 10, October 1990.

[11] Millikin, M., "Networks and Objects," *ConneXions,* Volume 5, No. 10, October 1991.

[12] Chappell, D., "The OSF Distributed Management Environment," *ConneXions,* Volume 6, No. 10, October 1992.

[13] Chappell, D., "The OSF Distributed Computing Environment (DCE)," *ConneXions,* Volume 7, No. 3, March 1993.

[14] Rosenberry, Ward, David Kenney, and Gerry Fisher, *Understanding DCE,* O'Reilly & Associates, 1992, ISBN 1-56592-005-8.

[15] Shirley, John, *Guide to Writing DCE Applications,* O'Reilly & Associates, 1992, ISBN 1-56592-004-X.

JEFFREY I. SCHILLER has been MIT's network manager since the inception of the campus network in 1984. Prior to that, he maintained MIT's Multics timesharing system during the ARPANET TCP/IP conversion. He in an author of MIT's Kerberos authentication system. Mr. Schiller is an active member of IETF's Security Area Advisory Group (SAAG) and IRTF's Privacy and Security Research Group. He recently worked on the Internet privacy-enhanced mail standards and implementation. Mr. Schiller is a founding member of the steering committee for the New England Academic and Research Network (NEARnet). He received his bachelor's degree in electrical engineering from MIT. E-mail: `jis@mit.edu`

Learn more about the issues discussed in this article at INTEROP 93 August. Attend tutorial T28: "Securing Your Network: Knowledge, Tools and Technology" (Monday and Tuesday). Also, Mr. Schiller will be speaking in session S72: "In Perspective: Security," on Friday from 1:30 to 3:00pm.

The Internet as Service Provider

by Peter Deutsch, Bunyip Information Systems

Introduction

There was a time not so long ago when the Internet was primarily defined in terms of its underlying technologies. In describing the net, implementors talked mostly of routers, subnets and protocol suites. The most important task for developers was to spread connectivity and improve reliability and few networking specialists were working on tools to provide higher level functionality. The actual work done over the Internet was seen as the business of users, not the architects of the Internet itself and there was little development work going on to produce tools or services for this new work environment.

It can be argued that this is now changing. The past two years has seen a tremendous growth in the range and availability of practical new tools for finding and accessing resources, and more and more administrators are taking up these tools to offer interesting and useful collections of information and other resources.

Where before users saw bits and bytes, individual hosts and specific service providers, they now see information and resources provided by "the Internet" itself. We are perhaps at last reaching a point where we can begin to regularly hide the underlying technology and concentrate on using the Internet to help with specific user tasks.

Of course, the Internet can still be a confusing place for new-comers. The range of interesting projects and prototype tools available seem to grow daily, yet the available documentation and information describing such tools remains scarce. If newcomers are to make sense of the bewildering range of tools and services available then a simple "Internet User's Tool Catalogue" seems to be in order.

In this article we will try to summarize the defining characteristics and distinguishing features of some of the more interesting and useful information delivery tools now available on the Internet. We hope that in summarizing such tools we will help newcomers better understand how to get started using the Internet.

Bear in mind that this article is *not* intended to be a complete user's manual for the entire Internet, nor is it intended to be a complete guide to any one tool. Instead, we try to provide a brief but comprehensive classified directory of tools, summarizing as briefly as possible what is available to you as a user of the Internet. The objective is to show how each project fits into the global collection of tools now available to you as an Internet user, not make you an instant expert on any one tool. Pointers to additional information are provided at the end of this article to allow you to find out more about specific projects that catch your fancy.

What kinds of services are out there?

The number of Internet information tools is large, and growing fast but despite first appearances, there *is* order in the seeming chaos. Certain techniques reappear regularly and many seemingly different tools perform similar tasks. Given this, it is possible to offer a relatively simple classification of projects that will encompass most of the existing tools and services. Bear in mind that the classification presented here is only one possible such ordering. Further, despite anything someone may write, you as a user may choose to use a particular tool for any task you wish. The goal is not to define what you can do with a particular tool, but to help explain how each tool might fit into your own particular Internet toolbox. Choosing the appropriate tool for your task, and learning to apply it well, is still a job left to your own energy, drive and imagination.

We now examine each type of service in turn.

The basics

The first applications on the net to find wide-spread application were the "big three"—Telnet, FTP and e-mail. For those who have not yet discovered them, here is a brief overview of each.

Telnet

Telnet is the name of a basic *protocol* that allows interactive communications between hosts. It is also the name of a *program* available on many machines that use this protocol to allow users to log onto other machines to work. With telnet, you can log onto and work on other hosts across the net.

The *rlogin* command, one of the so-called "Berkeley *r*-commands," is another program that allows you to log onto other machines and is available on machines whose operating system was derived from the Berkeley versions of UNIX. The *rlogin* command has the advantage that it passes a little bit more information about your environment on the calling machine to your target host than telnet does, and it allows the use of `.rhosts` files to allow you to move from machine to machine without using passwords.

On the other hand, you can usually assume that telnet will be available on any host that allows remote login capability. This is not true for *rlogin,* so it's definitely worth becoming familiar with the basics of telnet as you first get acquainted with the Internet.

FTP

FTP is the *File Transfer Protocol.* This is the standard protocol on the Internet for transferring files from one machine to another. It is also the name of the *program* on many machines that is used for accessing this protocol when you want to send and receive files between hosts.

Anonymous FTP is a simple but powerful extension to the basic FTP mechanism used to create and share public archives with others on the Internet. Those sites which wish to share such information with the community create a special account for a pseudo-user named "anonymous" on their machine. On many machines this account has no password, while on others a well-known string, such as your e-mail address, is used.

The anonymous account is almost always set up to only allow users to connect using the ftp command, conventional logins are usually disabled for this account. The anonymous account is also usually limited to fetching files from the archive (although in some cases, depositing files into a so-called "incoming directory" is also allowed). A huge amount of information is currently available through anonymous FTP. This includes public domain or freely available software, abstracts and full papers published by many researchers, public domain books and short stories, Internet standards and information documents and much more. One estimate put the combined set of anonymous FTP archives at over 2.5 million files on over 1,500 hosts.

The *archie* service is used to index the entire set of anonymous FTP archives and can be used to locate specific files available through anonymous FTP. For information on *archie,* see the section "Indexing services," below.

E-mail

E-mail or Electronic mail is an automated message processing service that allows you to exchange messages with other users. To send a message you will need the user's "e-mail address," and an electronic mail program for creating and processing the message.

continued on next page

The Internet as Service Provider *(continued)*

There are a wide variety of e-mail programs in use on the Internet, but most rely on one of a few basic underlying transport mechanisms to actually deliver the mail. Users usually compose their messages using a mail user agent or text editor and then use the user agent to pass the mail to a mail delivery agent for transport to the final destination. Common mail user agents include the Berkeley *mail* program (available on most UNIX-based systems) as well as such public domain or freely available mail programs as *Pine, mush* and many others.

The most common e-mail delivery mechanism on the Internet is SMTP (the *Simple Mail Transport Protocol*), which is a protocol used for transferring messages between different hosts. MIME (*Multipurpose Internet Mail Extensions*) is a recent extension to basic Internet mail that permits the transfer of sounds, graphics and other multimedia mail messages.

The X.400 mail protocol is another e-mail transport standard that finds use on the Internet. Although the two protocols cannot communicate directly, gateways between SMTP based mail delivery agents and X.400 mail delivery agents exist.

The next set of services we examine may be thought of as a set of extensions to basic e-mail. Mailing lists generalize the concept of sending a message to another user by allowing users to automatically send messages to an entire group of recipients in one step by sending mail to a single address for "explosion" or automatic distribution. Mailing lists are usually set up to allow people with a shared interest to communicate.

Bulletin boards

Bulletin Board services carry this idea one step further, allowing users to send out messages that can be accessed by any number of others using similar bulletin board systems through the world. When you send a message to a bulletin board there is no way to know in advance who will read your message or where it will go. Where mailing lists are a form of "semi-private" mail, bulletin boards are true public message systems.

Perhaps the best known bulletin board service on the Internet is *USENET,* although there are others, including gateways to commercial bulletin boards, such as Compuserve.

Mailing lists

There are several different ways to generate and maintain *mailing lists* on the Internet. On UNIX-based systems this can be done through an "alias" file, in which a list of valid e-mail addresses is associated with the name of a particular mailing list (known as the "mail alias"). Users of such a list simply send mail to the alias address and the appropriate mail system then expands the message, sending a copy of the message to each address on the list.

LISTSERV is a service for maintaining automated mailing lists that was first developed in the mainframe (BITNET) environment. In addition to handling the simple expansion of mail aliases, LISTSERVs are capable of responding to a number of commands sent via e-mail to the LISTSERV itself. These include commands to subscribe or unsubscribe an address from a mailing list, obtain a list of valid mailing lists, and more.

Mailbase is an enhanced LISTSERV-like mail service developed for the UNIX environment developed at the University of Newcastle upon Tyne in England.

Interactive information delivery services

Although both e-mail and anonymous FTP can be used to share information across the Internet, neither system allows simple browsing and neither is particularly easy for the new-comer to learn. Gopher and the World Wide Web (or W3) are two recent development projects that take aim at the problem of making it easier to distribute information in an Internet environment. Both allow the user to browse information across the net without the necessity of logging in or knowing in advance where to look for information.

Gopher

The *Gopher* system [3] offers the information to be served as a simple hierarchal system of menus and files. Although gateways to additional services are also provided, this is usually done transparently to the user. Operators of the Gopher system can easily offer a hierarchal directory of information across the Internet and the information served can be accessed using a wide variety of public domain or freely available client programs.

The Gopher project was first developed at the University of Minnesota and began life as a simple tool for offering user support information to their user community. It has now developed into a powerful distributed system for offering information across the net and finds widespread use as a *Campus-Wide information System* (CWIS).

The simplicity of the underlying Gopher protocol makes creating new Gopher clients a relatively simple task and a great deal of work has been done in this area. Gopher clients are available for virtually all of the major operating systems in use on the Internet.

World Wide Web

Where Gopher allows the user to browse menus and select individual documents for browsing, the *World Wide Web* project [4] allows the use of hypertext links from one document to another to allow the user to follow links from within one document to another. The W3 project also allows the user to annotate documents (again using hypertext links) and allows the user to see the entire collection of information available within the system as a single large mesh of information, freely available for browsing, searching and access.

Both projects offer gateways to additional collections of information and a range of free clients for many environments.

Directory Services

Directory Service tools are intended to provide a lookup service for locating information about users, services or service providers. The general topic of Directory Services is often divided into more specific *White Pages* and *Yellow Pages* services. For the purposes of this article, we say that a White Pages service provides information about individual *users,* while a Yellow Pages service provides corresponding information about *services* and service providers.

As an example, a White Pages service might be used to locate the e-mail address of a someone, given their name and organization, while a Yellow Pages service could be used to locate a particular online library catalogue or a particular file archive site.

WHOIS

There are now several different types of Directory services under development on the Internet. One of the first such services deployed was the *WHOIS* service, a bare-bones user directory originally created to track key network contacts for the early DARPA Internet. The WHOIS service is basically a very simple White Pages service, although some services contact information is included in the prototypical WHOIS database maintained by the DDN Network Information Center on the host `nic.ddn.mil`.

continued on next page

The Internet as Service Provider *(continued)*

A number of other sites now operate additional WHOIS servers, with a range of extensions and enhancements to the basic WHOIS model. WHOIS enjoys the advantage that is it a simple, no-frills service, and the software to contact a WHOIS server is available on a large majority of Internet-connected hosts. Work is now underway on what is known as the *WHOIS++* project (pronounce "WHOIS-plus-plus") to standardize a set of simple extensions to the WHOIS service to increase its functionality and ease of use.

X.500

The *X.500 Directory Service* [5] is a much more ambitious Directory project that has been under development for a number of years. Pilot implementations are now available from a number of sites and commercial versions are also available. Proponents have suggested that X.500 could also be used to provide a Yellow Pages service, although the pilot projects have concentrated on White Pages functionality.

Despite years of research and work, there is still no single Directory Service for the entire net. The biggest obstacle to such a service would appear to be the huge cost of setting up and maintaining the required databases which in turn leads to a classic "chicken and egg" syndrome. Lacking a large installed information base, few users demand suitable tools for accessing the service. Without a large user base, few sites invest time and energy to bring up or maintain their databases.

There are now over 100 sites operating their own WHOIS directories, and several hundred that have participated in the prototype X.500 research. Most of this combined effort has concentrated on White Pages services. It is to be hoped that as the range of other services grows, the demand for a functional Yellow Pages service will be the catalyst that finally pushes Directory Services forward.

Indexing services

There are now several Internet-based projects that aim to build up indexed catalogues of information for rapid searching and retrieval of information. The first such services aimed at providing network access to library card catalogues, with more recent projects offering technology to gather, index or catalogue other types of information.

Archie

The *archie* service [1] began as a simple project to catalogue the contents of hundreds of online file archives. The *archie* service gathers together the location information, name and other details for files and indexes them in a dedicated database. Users can then contact an archie server and search this database for needed files.

The follow-on to the initial *archie* service is now a commercially supported product of Bunyip Information Systems Inc. of Montreal and has been designed to gather and serve other useful collections of information. The latest version is being used to provide a prototype Yellow Pages service and directories of online library catalogues and e-mail mailing lists. Work continues on extending the *archie* service to provide additional collections of information.

The *archie* service is accessible through a range of access methods, including telnet, stand-alone client programs that run on your own machine, or via e-mail. The prototype *archie* service now tracks over 2,100,000 filenames on over 1,200 sites around the world. There are currently about a dozen *archie* servers around the world, with more on the way. For additional information on *archie*, send e-mail to `info@bunyip.com`.

PETER DEUTSCH holds a B.Sc in Mathematics and Computer Science and an M.Sc. in Computer Science from McGill University in Montreal. He has been involved in bringing the Internet to Canada since 1986 and is an active developer of new Internet information technologies. He is one of the creators of the *archie* information system and is active in the IETF standards-setting process, including efforts to develop Uniform Resource Identifiers and WHOIS++. He is President of Bunyip Information Systems Inc., a company specializing in Internet-based information services, and travels extensively lecturing on Internet information technology. `peterd@bunyip.com`

WAIS

The *Wide Area Information Services* (WAIS) [2] is a system for indexing and serving information in an Internet environment that offers a rapid search capability as well as "relevance feedback." This means that the results of one search may be used to successively refine future searches, thus making it easier to find what you are looking for. WAIS clients allow the user to specify searches as simple English-language queries, without any complicated logical expressions or command syntax.

The heart of the WAIS system is a set of WAIS servers that allow the operator to set up an index of each document (or resource) to be published onto the Internet. Such indices allow the server to rapidly locate keywords during searches. The matches found are ranked according to perceived relevance and an indicator returned to the user, who may choose to then view the document or further refine the search.

There are currently several hundred WAIS sources being served on the Internet. Public domain source code for WAIS servers and clients is for anonymous FTP from the machine `quake.think.com`, along with a master directory of the sources currently available through WAIS.

Online library catalogues

A large number of libraries make their computerized *library catalogues* available over the Internet. Most are available through telnet sessions (that is, you telnet to a specific address and login as a particular user). Some are also available through other tools, such as Gopher.

Without a functional Yellow Pages service finding a specific online library catalogue is still a difficult problem. One approach is to consult the document "Accessing On-Line Bibliographic Databases," compiled by Billy Barron of the University of North Texas (`billy@unt.edu`). This document lists the address and contact information for over 130 such catalogues and also contains useful information on the various types of interfaces found throughout the Internet. Copies of this document are available from a variety of anonymous FTP sites, although you should check the "last modified" date since such files do age.

Where do you go from here?

This completes our brief tour of existing Internet tools and services. You will obviously still have a number of unanswered questions, or perhaps just want to find out more about a particular tool. There are a number of places you can now turn to for more information. In particular, there are a number of books now available that contain more information on each of the systems described here. A list of these is included in RFC 1432 "Recent Internet Books."

References

[1] Deutsch, P. & Emtage, A., "The *archie* System: An Internet Electronic Directory Service," *ConneXions,* Volume 6, No. 2, February 1992.

[2] Kahle, B., "An Information System for Corporate Users: Wide Area Information Servers," *ConneXions,* Volume 5, No. 11, November 1991.

[3] McCahill, M., "The Internet Gopher: A Distributed Server Information System," *ConneXions,* Volume 6, No. 7, July 1992.

[4] Berners-Lee, T., "A Summary of the WorldWideWeb System," *ConneXions,* Volume 6, No. 7, July 1992.

[5] Benford, S., "Components of OSI: X.500 Directory Services," *ConneXions,* Volume 3, No. 6, June 1989.

Learn more in Session S31: "Resource Discovery," Thursday at 10:30am.

ATM for your internet—But When?

by Mark Laubach, Hewlett-Packard Laboratories

Introduction

Asynchronous Transfer Mode (ATM) represents an exciting new technology area for networking. ATM promises everything but just how real is it? This article summarizes the current state of ATM and enumerates a portion of the work that still needs to be done in order for ATM to become globally successful. The article then details current ATM testbed work at the Hewlett-Packard Laboratories in Palo Alto, California and summarizes our early experiences.

Good news and bad news

ATM does not escape the "good news" / "bad news" scenario. The good news is that ATM is fun, promising, and exciting technology. It offers the speed and performance that we need for the future; it promises the ability to handle aggregate traffic streams of differing *Quality Of Service* (QOS), e.g., mixing your real-time teleconference with your background file transfer; and ATM has the potential of providing global, infinitely scalable connectivity from local LANs to long haul links, from backbones to breadboxes. Yes, there are technology investigations in progress that are trying to bring ATM to your TV set top—imagine when it reaches the kitchen. The bad news is that ATM won't be able to achieve these promises for a sizable number of years —there is much work to be done.

To paraphrase the words of a colleague from XEROX PARC: "It would have been much better if we could have kept the lid on ATM for another year so we could have gotten a few things worked out."

Things that need to be worked out

ATM will be globally successful when the following is available across all implementations: signaling, congestion control, traffic management, multimedia support, internet interoperability, and security.

Signaling enables end-points (e.g., hosts, toasters, video game consoles) to talk to their network connection (switch) and for switches to talk to switches. The signaling protocol is called Q.93B and is currently under development. At the time of writing, the first release of the ATM signaling specification is being taken from *Draft* status to *Implementation Reference* status in the ATM Forum [1]. Q.93B provides at a minimum, the same equivalence as the DTMF pad on your voice phone—it allows the user to place a call into the network (call setup) via a standard mechanism. Q.93B also provides user information elements that allow terminals (hosts) to signal other aspects of a connection such as quality of service requirements (peak bandwidth, mean bandwidth, best effort, etc) and end-user application needs such as maximum transmission unit size, what application to bind to the connection (e.g., IP, ARP, a TCP port, etc). Q.93B for the most part is a paper specification at this time. The first reference implementations should appear sometime this fall. It will be more than a year until Q.93B is implemented in the majority of vendor ATM switches and longer still until the internet community implements wide-scale services that make full use of it.

Congestion Control enables ATM switches to throttle incoming traffic when resources are in short supply. This is also called flow control. Current implementations of ATM switches benefit from being very fast as compared to the incoming traffic. This allows cost and technology shortcuts to be made as the switch will always be able to handle offered load. However, as ATM is used more heavily (faster hosts, many more hosts, and demanding applications) switches will saturate and cells will be dropped.

Congestion control are the mechanisms an ATM switch will use to both manage its resources so that it never runs "dry" and that the switch uses to communicate to the upstream nodes in order to push back or throttle traffic. The goal of congestion control is that internal switch buffers are efficiently used and that cells are never dropped. This area is currently a research topic. It will be several years before the community sees any ATM congestion control standards implemented in products.

Traffic Management is the mechanism and policies each ATM node uses to decide how and when it schedules cells for transmission. Traffic management includes aspects of fair queue management and a policy-based scheduling mechanism that enforces quality of service contracts. Traffic management mechanisms can be combined with congestion control mechanisms to provide robust and flexible resource management within an ATM switch network. Current research is examining more comprehensive traffic management algorithms based on weighted fair queuing that may alleviate the need for a separate flow control mechanism. Needless to say there is much research yet to be done in traffic management and it will be at least a year before the community sees the results from the research implementations.

Multimedia Support is the ability of each node of an ATM network to handle aggregate streams of traffic, each with a different quality of service contract. For example, an ATM switch may be handling several video and audio streams, FTP file transfers, AFS file sharing, etc., each with their own QOS contract. Proper switch management will enforce that the bursty nature of the FTP and AFS transfers will not interfere with the timely transmission of the video and audio streams. Note that this is more than just a traffic management issue as ATM nodes must have the proper hardware mechanisms so that high priority traffic is not placed in the same queue behind lower priority traffic (*serialization*). This requirement implies multiple hardware queuing mechanisms, e.g., separate output FIFOs for high-priority and low-priority traffic in an output buffered switch. It will be several years before we see hardware support for multimedia in place in the majority of vendor switches.

Internet Interoperability is very basically how the Internet uses ATM to get from place to place in a seamless fashion. The *Internet Engineering Task Force* (IETF) "IP-over-ATM" Working Group is currently drafting specifications for the implementation of IP and ARP over ATM [2]. Initial proposed standards will be out soon for specifying how the existing IP and ARP protocols will interoperate over ATM. The Internet community will not see global IP interoperability until the ATM vendor community implements the proposed standards.

Security: Global ATM connectivity implies that virtual circuits could be established between any set of end-stations. Imagine the workstation or PC at your desk accepting a switched virtual circuit from anywhere else in the world. Do you know who or what is at the other end of that connection? Global ATM will need to have secure authentication capabilities built in to the call setup process. Currently there is no activity in the IETF or the ATM Forum on the specification of secure call setup.

This is not the exhaustive list of things that need to be worked out. There are other areas in ATM that need further development: addressing, switch management, switch-to-switch signaling, multicast, private network-to-network interface standards, and more. This article only discusses a subset of the work in progress.

continued on next page

ATM for your internet *(continued)*

There is much work going on in ANSI, ITU-TSS (CCITT), the IETF, and the ATM Forum to develop standards that will improve the offering and capabilities of the basic ATM technology. We are seeing proprietary and experimental solutions that are exploring these issues and providing early implementations and experience.

Deployment

The acceptance of ATM into the internet community will take a few years until the "things to be worked out" are resolved, standardized, and widely implemented. The deployment of ATM during this time of evolution will initially follow traditional broadcast and firewall boundaries because:

- Administrators will tend to use the same models to replace local area network segments (broadcast LAN segment out, ATM LAN segment in) and the "wires" between routing elements, i.e., leased lines will migrate to ATM SONET permanent virtual circuits.
- Policy administration practices rely on the security, routing, and filtering capability of border gateways: i.e., firewalls. ATM virtual circuits will not be allowed to back-door these mechanisms. Additionally, current security practices rely on the slow turn around time to install and verify end-points with leased lines.
- Interoperability standards, such as IP over ATM will take some time to be implemented in vendor products.

So, the Virtual Circuit stops here, and there, and doesn't go everywhere yet, but it will.

Caveats and recommendations

Despite the shortcomings identified thus far, ATM is still a viable technology for (re)building your local area networks. However, it is best to "know before you go" and understand the following caveats:

- Keep your current internet working smoothly as you integrate ATM local LAN switches and backbones, i.e., deploy ATM as seamlessly as possible.
- Deploy ATM anywhere within your "fence." Non-local ATM virtual circuits should terminate on the non-local side of your firewalls, unless you really want to accept the risk of VCs getting around your IP security fences.
- Get your current network administrators involved now in the planning from the ground up if they are not involved already.
- Be prepared to roll your ATM equipment within the next few years as developing standards and performance improvements may require new hardware.
- Go have fun!

ATM at HPLabs

At HPLabs, Palo Alto we are actively involved in ATM technology research. At the core of our ATM activities is the construction of an ATM testbed. It is fundamental to our efforts that we understand ATM from the ground up. Specifically, we are having fun by:

- Deploying ATM as a local area network replacement in our laboratory and workgroup scenarios. This includes the preparation necessary for installing a well thought out fiber plan. Our local network will be implemented in a seamless fashion to allow IP interoperability.
- Participating in several Bay Area ATM testbed opportunities: the *Sprint Broadband MAN* trial, the *Bay Area Gigabit Testbed* (BAGNET), and the *Smart Valley* network.

- Using our Sprint connections, we will be experimenting with geographical ATM connectivity between us and our HP Cupertino, California site.
- Investigating applications that benefit from having ATM as the physical layer: high bandwidth applications, guaranteed quality of service, obligatory video teleconferencing, etc.
- Working in the standards group to help guide the future of ATM: ATM Forum and the Internet Engineering Task Force (IETF).

HP Labs has actually been waiting for the next generation backbone technology for years now. It has turned out that we will be just deploying ATM as it now appears more viable and flexible than FDDI. We have spend considerable deliberate time preplanning fiber installations throughout the Palo Alto campus. We have deployed both single mode and multimode over the past couple years between all buildings as appropriate. We even have multiple path connectivity planned so that we can avert catastrophic "backhoe" or earthquake failure for our critical paths. We tend to be progressive in the investigation of new technology. Where we find ourselves successful, we try to leverage our experiences over to our corporate networking group.

HPLabs and the Sprint Broadband MAN trial

HP Labs is participating with Sprint Research Labs in Burlingame, California in the study of the deployment of SONET and ATM over a metropolitan network (see Figure 1). Our studies over the next two years will include using ATM over SONET, geographical connectivity of ATM to our sister HP site in Cupertino, California, and in the use of applications that will benefit from the higher speeds and low latency of ATM networks: e.g., video conferencing, file sharing, etc. We will be initially using OC3 (155 Mbps) connections between our switches and the Sprint SONET network. OC12 may be available before the two-year experimental period concludes.

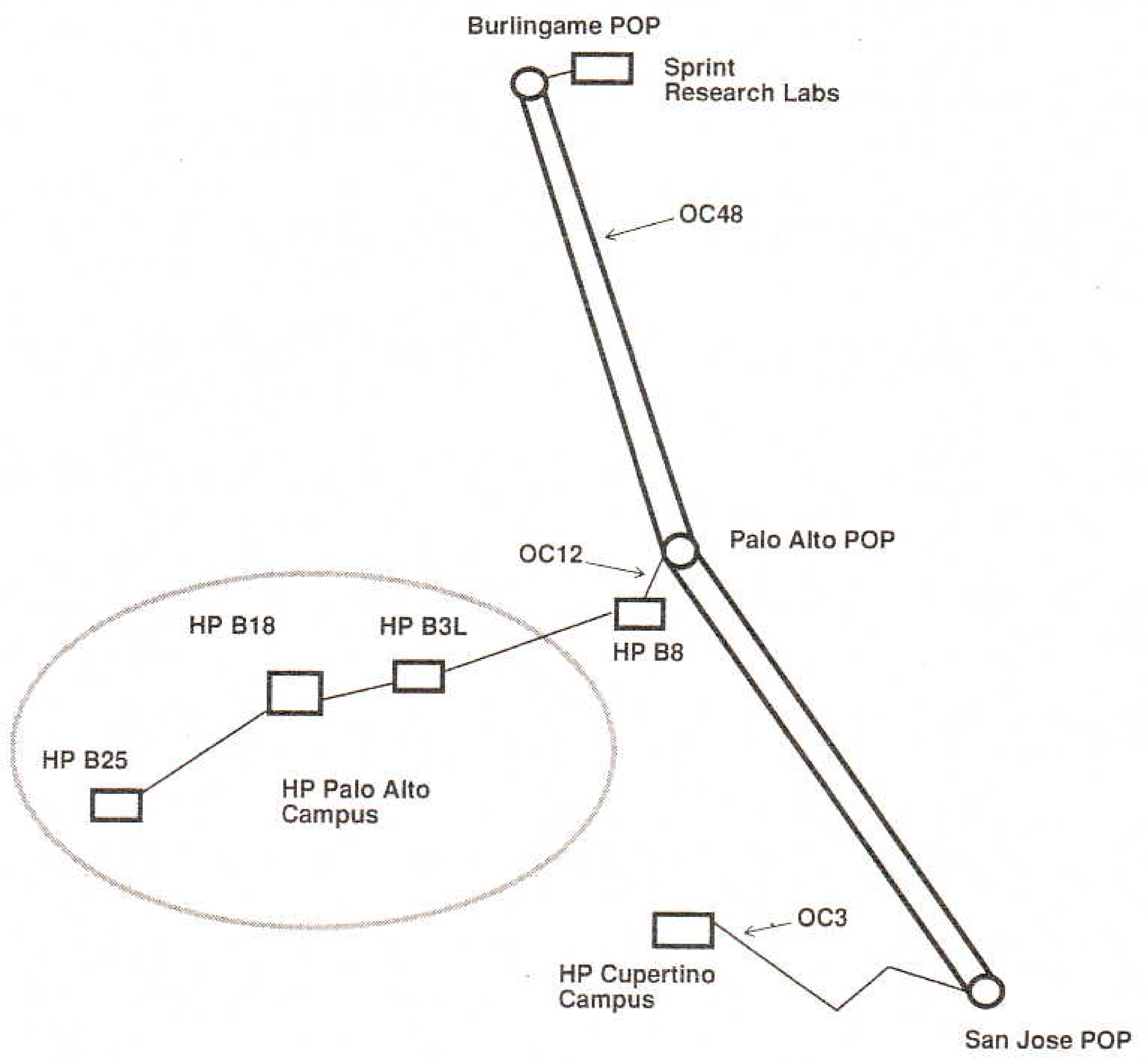


Figure 1: Sprint Broadband MAN trial

Inside our local ATM testbed, we are using Fore Systems Inc. ASX-100 ATM switches configured with four OC3 ports, four 140 Mbps TAXI ports, and four 100 Mbps TAXI ports. Additionally, we will be using an experimental ATM switch with six 100 Mbps TAXI ports as an extension of the Fore switch. Our hosts will be HP 9000 Series 700 HP-UX workstations. All ATM host interfaces will be EISA bus based. We will dual home all ATM hosts to an Ethernet which will be connected on the outside of our corporate IP firewall. Strict security measures will be maintained on these machines. See figure 2.

continued on next page

MARK LAUBACH holds a BEE. and M.Sc. from the University of Delaware. He is currently an Engineer/Scientist in the Media Technology Lab at the Hewlett-Packard Laboratories. Mark's current research includes multimedia networking, high-speed "gigabit" networking, and the construction of ATM testbeds. He has been with the Hewlett-Packard Company for 13 years. Mark is a member of the IETF and is the chair of the IP-over-ATM Working Group. He is the author of the Internet Draft "Classical IP and ARP Over ATM." He is a past member of the CSNET Executive Committee and a past member of the board of trustees for the Corporation for Research and Educational Networking (CREN, CSNET+BITNET).

ATM for your internet *(continued)*

Our interoperability goals for our ATM testbed are:

- IP over ATM is our interoperability standard
- SNMP Management of all ATM switches and host interfaces
- Permanent Virtual Circuits initially for "public" connections, and Switched Virtual Circuit experiments using Fore's SPAN protocol for local and HP geographic connections
- Experiment with Q.93B when available
- Firewalls kept in IP (and higher), possible filters in the HP-UX ATM driver
- Fiber to the desktop

Our application experiments will include:

- IP applications that run fast: NFS, AFS, e-mail, X, etc.
- Video over ATM in HP-UX:
  - FORE EISA ATM host interface
  - HP EISA ATM host interface
  - MediaMagic EISA Video Card (with JPEG)
- We're thinking more up.......

Learn more in Session S30: "ATM Testbeds," Thursday at 10:30am.

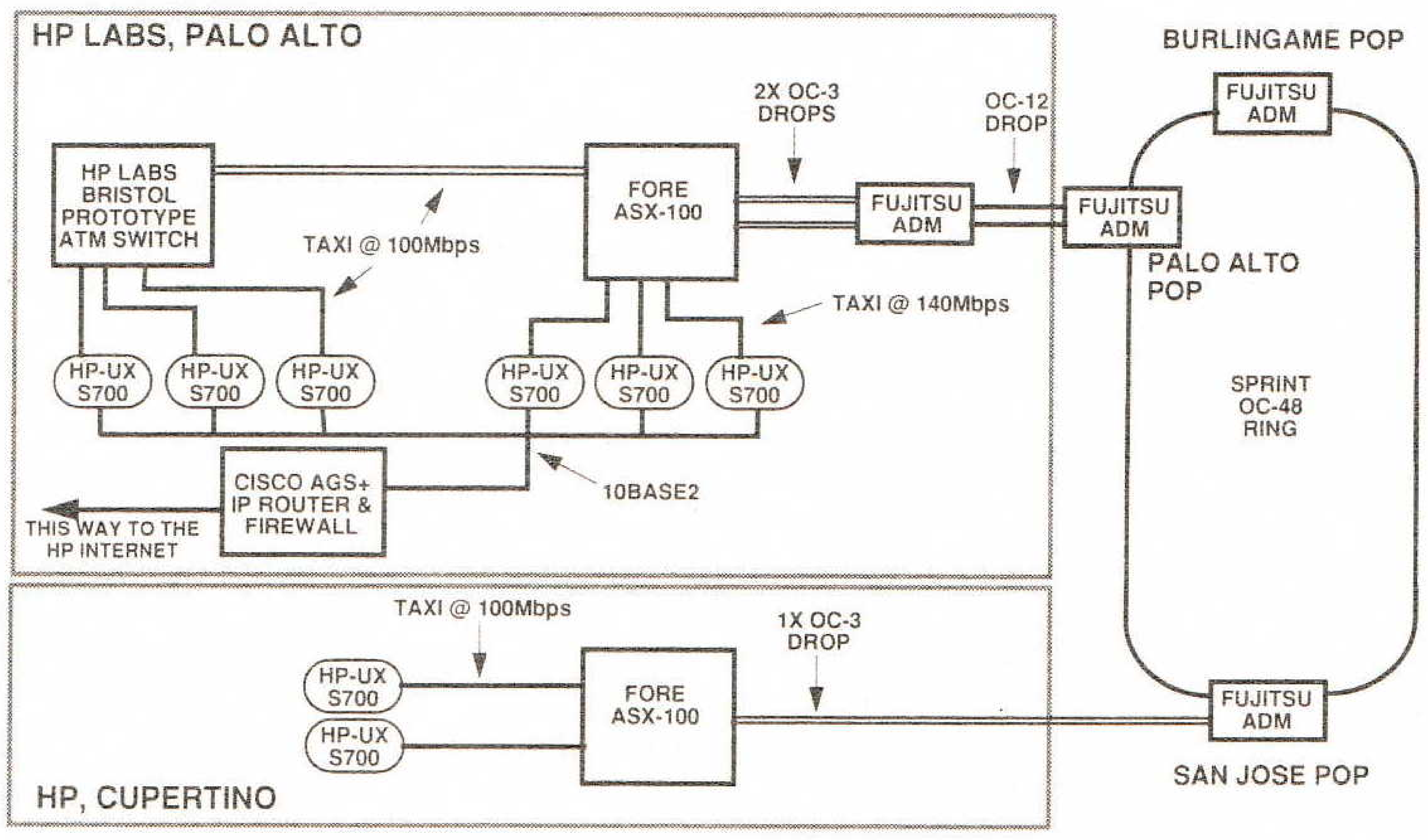


Figure 2: HPL ATM Testbed configuration

HPLabs ATM experience summary

We have found that ATM is indeed "fun" to use and we want to start using it more widely in our local networks. Our networking systems teams is coming up to speed quickly and have been able to share the load in the care and feeding of our ATM switch—they are in a position to deploy more fully. We believe that any experience you can get with ATM technology now will yield an abundance of valuable lessons. We've also found that the lack of a standard, widely deployed, signaling protocol (i.e., Q.93B) is a severe hindrance in the deployment of ATM. The hand configuring of PVCs wastes time and introduces replication errors as operations need to be repeated on all hosts.

In summary, ATM is young and the technology and standards are quickly evolving. ATM promises to be the answer for building high speed, seamless, globally deployed networks—it will just take a little while to get here. In the mean time, use it and have fun.

References

[1] ATM Forum, "User Network Interface (UNI) Specification Version 3.0," August, 1993.

[2] Heinänen, Juha, "Multiprotocol Encapsulation over ATM," RFC 1483, March, 1993.

GateD and the GateD Consortium

by Martyne Hallgren and Jeff Honig, Cornell University

Introduction

The Cornell GateDaemon project produces *GateD* (pronounced "gate-dee"), a modular software program which supports multiple routing protocols and protocol families. GateD performs routing functions through the use of several common routing protocols. Originally developed for the initial NSFNET backbone, GateD was designed to listen to the different routing protocols and choose the best route to a given network.

GateD is now used by hundreds of organizations worldwide. It is currently being deployed in the NSFNET/ANSnet (T3) backbone and has been running on the CA*NET (Canada) backbone for over a year. The software is available by anonymous FTP for individual use. It is licensed for redistribution, free of charge, to anyone, as long as they agree to return any enhancements to the project for redistribution to all users. This collaboration has been overwhelmingly successful. Users continually contribute changes and new protocol implementations for inclusion in the most current distribution.

The GateD software can be used to bootstrap and support the production of interoperable networking products. It gives organizations access to leading edge network research and implementations without tremendous startup investments. The software is tested extensively by users throughout the Internet community before and while the code is in production. Contributions of protocol implementations and enhancements are integrated into the distribution package.

The GateDaemon Consortium

The Cornell *GateDaemon Consortium* was conceived to foster the development of the project. The Consortium provides a framework to strengthen and expand the already successful collaborations currently in place. The GateDaemon Project has been supported mainly by the National Science Foundation and Cornell University since 1988.

Cornell's goal is to provide a "state-of-the-art" software platform to support the most current routing protocols for the Internet community. The demands from the community continue to outpace the resources currently available to the project. Instead of adopting a licensing or royalty fee structure, the Consortium will use membership fees to augment grant and proposal funds and provide resources to respond to the rapidly changing needs of the community. Membership in the Consortium benefits both operators and developers alike, by making excellent routing protocol implementations available throughout the Internet.

In addition to the assurance that GateD software continues to be freely available to the Internet community, the Consortium provides direct benefits to its members including in-depth technical seminars on new features and capabilities, discussions on new features and goal setting with the GateD developers, and improved communications about the project and Consortium activities.

History

In 1986, NSFNET and the emergence of several regional networks created a much more complex system than ever before. Instead of a single path, as could be found in ARPANET, there were several paths available for use. While work was ongoing to find a long term solution, network operators needed an immediate solution to provide dynamic routing among the mid-level networks, campuses, and the NSFNET backbone while taking advantage of the added redundancy now available.

continued on next page

GateD and the GateD Consortium *(continued)*

Given a short development and deployment schedule, the development of GateD began in August of 1986 at Cornell University by Bill Nesheim and Mark Fedor. It was initially designed to run under Berkeley UNIX and implemented the three most common routing protocols in the Internet at that time: RIP, HELLO, and EGP. The first version was made available by early November 1986 with the valuable help from a number of test sites. One of the earliest contributions came from the folks at the University of Maryland, who contributed much of the initial HELLO modules.

Since that time, GateD has become the de facto reference implementation of routing protocols and a popular platform for routing protocol development. As it was in its handling of the original problem it was meant to solve, GateD is at the forefront of delivering mechanisms for managing the growing complexity of the Internet.

GateD implementation Release 3.0

GateD is a modular program consisting of a *core,* a *routing database,* and *protocol modules.* The protocols are independent of each other as much as possible. Selective compilation has been implemented to allow network administrators only to compile the protocols they need for their site, thus reducing the size of the program.

It is more configurable than some commercial routers, allowing minimal configuration for simple cases and ease of use when configuring more advanced features. Dynamic reconfiguration can be affected at any time by sending a signal to the running program. The network administrator can specify a preference level for each combination for routing information to be imported. The decision on which route to use is based on the preference level the administrator has chosen.

GateD has been ported to many flavors of UNIX as well as non-UNIX platforms. Originally based on BSD UNIX, the current routing database interfaces take advantage of the capabilities in the "4.3 RENO" version of BSD UNIX in the area of routing, especially the routing socket and variable-length subnet mask capabilities. These new capabilities will provide the foundation for supporting multiple, highly disparate protocol families with ease. It also has front-ends for almost all system functions, including memory management, to make the protocol software as system-independent as possible.

Protocol support

GateD supports both distance vector protocols and link-state protocols. Release 3 currently supports the following protocols: BGP 2/3, EGP 2, OSPF Version 2, RIP 1/2 and DCN HELLO. Similar protocols e.g., BGP and EGP have similar configuration syntax for ease of use.

As the internetworking environments have become more complex, it is more and more important to provide debugging tools to help administrators. To aid them, tracing capabilities have been increased, especially for OSPF. In addition, on most systems it is possible to dump the entire state of GateD, including the routing database, without impacting the protocols which need to respond rapidly to changes.

The services offered to protocol modules are considerable. Both unlimited interval timers and single-event timers, packet arrival indications, and other events are provided to the protocol modules by core services.

GateD currently supports EGP 2 as described in RFC 904. This is a complete implementation of the specification. Optimizations for MILNET have been added, especially configurable parameters for maximum packet size and HELLO and POLL timers.

OSPF Version 2 as described in RFC 1247 has been implemented as well as the OSPF MIB described in RFC 1253. Unlike other protocols, reconfiguration of OSPF currently requires stopping and restarting the protocol. This implementation utilizes local-wire IP multicast on systems that support it.

Both RIP 1 as described in RFC 1058 and RIP 2 as described in RFC 1388 have been implemented. RIP 1 features include support for split horizon without poison reverse (see footnote at end of article) to reduce routing overhead and implements graceful shutdowns to avoid black holes. RIP 2 uses IP multicast where available. Variable subnets and next hops are fully supported.

DCN HELLO also supports split horizon without poison reverse to reduce routing overhead. Currently, plans are being developed to retire this protocol from the GateD distribution.

New features in Release 3.0

In GateD Release 2.0, the knowledge of system interfaces was almost static. It would only acknowledge the presence of interfaces available when it was started. It did not notice interfaces that became available at a later time nor did it notice "pop-up" interfaces such as those used by some SLIP and PPP implementations, that are created dynamically by the kernel. In Release 3.0, dynamic interfaces are supported such that they can be added or deleted and indicated as being up or down. Changes to several parameters such as address, netmask, and metric are recognized and handled properly. There are several levels of interface references in the configuration language; interfaces may be identified using wildcards, interface type name, interface name or address. Where available, routing socket interface status messages are supported.

SNMP is now supported via the SMUX protocol. It has been developed using ISODE 7.0 with SNMP UPGRADE 1 because this is still freely available. SNMP supports traps and some writeable variables. A stripped-down distribution is available in the GateD archive files, but it is still large.

The routing table for GateD has been improved in Release 3.0 It is now event driven and steady-state processing is minimal. Policies are evaluated by protocols only when changes occur. Protocols will flag routes when they are announced and the routes will not release until the flag is reset. This allows holddowns to be implemented by the protocols, currently RIP and HELLO, that require them while not restricting protocols that do not require them.

The routing table is policy independent. Many protocols, specifically OSPF, EGP, and BGP, retain all routes and reconfiguration results with minimal impact to the routing system. Routes are selected by preference. The active route is chosen by the lowest preference. Preferences are configurable at many levels.

Variable subnet masks and routing sockets are only supported with 4.3 BSD RENO or later. There are bugs in versions before 4.4 BSD, but there are fixes available for RENO and later. The routing socket allows improved synchronization of the kernel routing table and indicates changes in interface status.

GateD futures

The next versions of GateD will include significant enhancements in functionality. One of the most eagerly awaited features is support of BGP 4. The BGP 4 implementation will include classless (CIDR) addressing and IDRP compatible AS paths.

continued on next page

GateD and the GateD Consortium *(continued)*

Timers have been added to dampen incoming changes to smooth transients from a noisy neighbor. An outgoing change delay can be used to allow an internal gateway protocol to converge.

Release 3.1 will support route aggregation. Route aggregation is used by *Classless Inter Domain Routing* (CIDR) to combat the explosive growth in the size of the routing tables. Route aggregation is also a generalization of the existing practice of announcing a route to a network at the border of a subnetted network, a generalization that is frequently wrong.

Several protocols support multiple metrics. A future version will distinguish values such as unreachable or N/A and will interpret the metric type to facilitate metric translation. Tracing will be enhanced such that it will have the ability to focus on specific groups, neighbors (BGP), or interfaces (OSPF). Tracing flags will be set per task and have a command line interface. The protocol modules will continue to be enhanced as more features are supported in each.

GateD collaborations

The GateDaemon project continues to rely on collaborations with the Internet community. In addition to a large number of individuals who work with the Cornell developers on testing new code and supplying bug fixes, several individuals are working on specific features which will eventually be included in the GateD distribution.

Steve Heimlich from IBM Corp. is working on an implementation of integrated IS–IS. Integrated IS–IS is an ISO interior gateway protocol that is also capable of propagating IP routing information. Steve's version is based on the DLS Associates' IS–IS and runs over ISO as well as IP. The *Inter-Domain Routing Protocol* (IDRP) is being led by Sue Hares from Merit Inc. IDRP is an ISO exterior gateway protocol which is capable of propagating IP routing information and also implements *Quality of Service* (QoS)

Jim Beers from Cornell has started to implement *Distance Vector Multicast Routing Protocol* (DVMRP), an inter-domain multicast routing protocol. This is the first step toward supporting multicast routing by a number of protocols in an integrated fashion.

Obtaining GateD

GateD source may be obtained using anonymous FTP, Gopher, or electronic mail. (See next page). Those organizations wishing to redistribute GateD Release 3.0 must sign a current redistribution license. (Note: This is a different license from Release 2.0).

Membership

The Consortium is open to universities, industry, and other organizations in any country with an interest in the Internet or in internetwork routing protocols. Membership, while strongly encouraged, is not, at present, a prerequisite for a redistribution license of the GateD software.

There are currently four levels of membership:

- Sponsor
- Associate
- Affiliate
- Academic or Startup

The membership level corresponds to the amount of participation an organization wishes to support. Membership is yearly.

JEFFREY C. HONIG has been the chief software developer for GateD since 1987. He has been involved in networking since 1982 on the campus, regional, and national level. He is an active participant in the design, specification, and implementation of routing protocols as a member of IETF working groups. He is also a consultant on engineering routing interfaces among Internet routing domains. He can be reached as: `jch@nr-tech.cit.cornell.edu`

For more information

Information on licenses and the consortium may be directed to the electronic mail address or postal address listed below:

GateDaemon Consortium
Cornell University
143 Caldwell Hall
Ithaca, NY 14853-2602
Attn.: Martyne Hallgren
Phone: +1 607-254-8767
E-mail: `consortium-interest@gated.cornell.edu`

There will be a GateD BOF on Wednesday night at 7:30pm.

Source code is available via anonymous FTP from Internet host `gated.cornell.edu`. The relevant files are `pub/gated/gated-R3_0Beta_1.tar.Z` and `pub/lists/gated-people`.

To use Gopher, connect to `gated.cornell.edu` and look in the GateD distribution Directory for the Gated 3.0 Beta 1 Distribution.

Mailing List Archives: `gated-alpha@gated.cornell.edu` and `gated-people@gated.cornell.edu`.

The distribution can also be retrieved via electronic mail:

```
To: gated-people-archive@gated.cornell.edu
Subject:
send gated/gated-R3_0Beta_1.tar.Zus*
help
send Contents
```

Footnote

"Split horizon" is a scheme for avoiding problems caused by including routes in updates sent to the gateway from which they were learned. The "simple split horizon" scheme omits routes learned from one neighbor in updates sent to that neighbor. "Split horizon with poisoned reverse" includes such routes in updates, but sets their metrics to infinity. See RFC 1058.

References

[1] Honig, J., "GateDaemon," Presentation Notes, GateDaemon Consortium Briefing, May 1993.

[2] Brim, S., "GateD: A Request for Renewal of Support," Cornell Information Technologies, Cornell University, August 1992.

[3] Hallgren, M., "Announcing the GateDaemon Consortium," GateDaemon Update, May 1993.

[4] Fedor, M., "The Gateway Daemon," *NSF Network News,* March 1988.

[5] Mills, Dave, "Exterior Gateway Protocol Formal Specification," RFC 904, April 1984.

[6] Hedrick, C., "Routing Information Protocol," RFC 1058, June 1988.

[7] Moy, J., "OSPF Version 2," RFC 1247, July 1991.

[8] Baker, F., Coltun, R., "OSPF Version 2 Management Information Base," RFC 1253, August 1991.

[9] Malkin, G., "RIP Version 2," RFC 1388, January 1993.

[10] Hares, S., "Components of OSI: Inter-Domain Routing Protocol (IDRP)," *ConneXions,* Volume 6, No. 5, May 1992.

[11] *ConneXions,* Volume 3, No. 8, August 1989, "Special Issue: Internet Routing."

[12] *ConneXions,* Volume 5, No. 1, January 1991, "Special Issue: Inter-domain Routing."

MARTYNE M. HALLGREN is the principal architect of the GateDaemon Consortium. She has been active in data networking on a campus, regional, and national level since 1986. Since coming to Cornell, she has taken a leading role as a user advocate in IETF, FARNET, and ACM. She is founder and chairperson of the Networking Taskforce in ACM SIGUCCS, member of the IETF User Services Advisory Council, and chairperson of the Client and User Services Committee for FARNET. She received a B.A. in Computer Science from the University of Nebraska-Omaha and will complete an MBA from Cornell University in 1993. E-mail: `martyne@nr-tech.cit.cornell.edu`

Introduction to Routing

by David M. Piscitello, Bellcore and A. Lyman Chapin, BBN

[Ed.: This is an excerpt from *Open Systems Networking: TCP/IP and OSI,* Addison-Wesley Publishers, 1993.]

Introduction

"Rö´•ting" is what fans do at a football game, what pigs do for truffles under oak trees in the Vaucluse, and what nurserymen intent on propagation do to cuttings from plants. "Rou´•ting" is how one creates a beveled edge on a table top, or sends a corps of infantrymen into full-scale, disorganized retreat. Either pronunciation is correct for "routing," which refers to the process of discovering, selecting, and employing paths from one place to another (or to many others) in a network.

The British prefer the spelling *routeing,* presumably to distinguish what happens in networks from what happened to the British in New Orleans in 1814. Since the *Oxford English Dictionary* is much heavier than any dictionary of American English, British English generally prevails in the documents produced by ISO and CCITT; wherefore, most of the international standards for routing protocols use the *routeing* spelling. Since this spelling would be unfamiliar to many readers, we use *routing* in this article, with apologies to our friends in the British Standards Institute.

A simple definition of routing is "learning how to get from here to there." (This is an application of the classic definition of a route by Shoch, tempered by the very practical observation by Radia Perlman that routes are both source and destination *dependent;* i.e., that knowing how to get there isn't enough, you have to know where you are starting from as well.) In some cases, the term *routing* is used in a very strict sense to refer *only* to the process of obtaining and distributing information ("learning"), but not to the process of using that information to actually get from one place to another (for which a different term, *forwarding,* is reserved). Since it is difficult to grasp the usefulness of information that is acquired but never used, we employ the term *routing* to refer in general to all the things that are done to discover and advertise paths from here to there and to actually move packets from here to there when necessary. The distinction between routing and forwarding is preserved in the formal discussion of the functions performed by OSI end systems and intermediate systems, in which context the distinction is meaningful.

Source Routing and "Hop-by-Hop" Routing

The routing operations of finding out how to get from here to there, and then actually getting from here to there, can be done in two (very different) basic ways. In *source routing,* all the information about how to get from here to there is first collected at the source ("here"), which puts it into the packets that it launches toward the destination ("there"). The job of the intervening network (with its collection of links and intermediate systems) is simply to read the routing information from the packets and act on it faithfully. In *hop-by-hop routing,* the source is not expected to have all the information about how to get from here to there; it is sufficient for the source to know only how to get to the "next hop" (perhaps an intermediate system to which it has a working link), and for that system to know how to get to the *next* hop, and so on until the destination is reached. The job of the intervening network in this case is more complicated; it has only the address of the destination (rather than a complete specification of the route by the source) with which to figure out the best "next hop" for each packet.

Example

Consider an example, in which "here" is your home in Hopkinton, MA, and "there" is Blueberry Hill Inn in Goshen, VT. If you sit down at home with a set of road maps and figure out exactly which roads and highways connect Hopkinton and Goshen, plotting the route you will follow along this road to that interchange to this junction (etc.), and then get in your car and actually drive along precisely that route to Blueberry Hill, you are performing source routing: if you were a packet, an ordered list of identifiers for links (roads) and intermediate systems (junctions and interchanges) would be encoded in your protocol header. If, on the other hand, you simply climb into your car and begin driving, stopping at every intersection to ask directions or examine the signposts, you are performing hop-by-hop routing: if you were a packet, the identification of your origin (Hopkinton) and final destination (Blueberry Hill) would be encoded in your protocol header. In the first case, your ability to actually get to Blueberry Hill depends on the accuracy of the maps you used and whether or not any of the roads you have selected are closed for repairs; in the second case, it depends on finding enough information at every intersection to enable you to pick the right road to follow to the next.

For the most part, routing in OSI and TCP/IP networks today is hop-by-hop. Source routing has recently emerged as an important component of a new set of routing capabilities (for both OSI and TCP/IP networks) that support complex *policies* governing the paths that packets are permitted to take when more than one organization owns or administers the equipment and facilities that intervene between "here" and "there."

Routing principles

The principal criterion of successful routing is, of course, *correctness* (you do in fact want to get to Blueberry Hill, not Cranberry Bog), but it is not the only criterion. You might prefer to take the most direct route (the one that takes the least time and uses the least fuel), the most reliable route (the one that is not likely to be closed by a heavy snowfall), the most scenic route (the one that follows pleasant country roads rather than busy highways), the least expensive route (the one that follows freeways rather than toll roads), or the safest route (the one that avoids the army's missile testing range). In its most general form, *optimal* routing involves forwarding a packet from source to destination using the "best" path. What constitutes the "best" path can, of course, become quite a complicated question, as this example shows; networks, like the highway system, have variable costs, transit restrictions, delay characteristics, and residual error rates, and all of these can be more or less important in the determination of what "best" means for a particular source and destination or for a particular packet.

The principal objective of an open systems routing architecture is not, therefore, to achieve "optimal" routing—such a thing does not exist in the abstract. Such an architecture must nevertheless be based on principles that account for what is happening in the real open systems world of today and tomorrow, in which computers are being connected to networks at a rate that more than doubles the number of systems connected to the worldwide (OSI and TCP/IP) Internet each year. These computers will be connected using a variety of local, metropolitan, and wide area networking technologies; the topology of interconnection will change as computers and the links between them are added and deleted; the networks will cross every conceivable national and international boundary; and the computers and networks will be administered by different organizations, both public and private, each of which may impose rules (policies) governing (and safeguarding) their use.

continued on next page

Introduction to Routing *(continued)*

Requirements

These observations suggest that an open systems routing architecture should:

- Scale well
- Support many different subnetwork types and multiple qualities of service
- Adapt to topology changes quickly and efficiently (i.e., with minimum overhead and complexity)
- Provide controls that facilitate the "safe" connection of multiple organizations

It is not likely that the manual administration of static routing tables (the earliest medium for the maintenance of internetwork routes, in which a complete set of fixed routes from each system to every other system was periodically—often no more frequently than once a week—loaded into a file on each system) will satisfy these objectives for a network connecting more than a few hundred systems. A routing scheme for a large-scale open systems network must be dynamic, adaptive, and decentralized; be capable of supporting multiple paths offering different types of service; and provide the means to establish trust, firewalls, and security across multiple administrations (see ISO/IEC TR 9575, the OSI Routing Framework).

OSI Routing Architecture

The architecture of routing in OSI is basically the same as the architecture of routing in other connectionless (datagram) networks, including TCP/IP. As usual, however, the conceptual framework and terminology of OSI are more highly elaborated than those of its roughly equivalent peers, and thus, it is the OSI routing architecture that gets the lion's share of attention here. Keep in mind that most of what is said about the OSI routing architecture applies to hop-by-hop connectionless open systems routing in general.

The OSI routing scheme consists of:

- A set of *routing protocols* that allow end systems and intermediate systems to collect and distribute the information necessary to determine routes
- A *routing information base* containing this information, from which routes between end systems can be computed. (Like a directory information base, the routing information base is an abstraction; it doesn't exist as a single entity. The routing information base can be thought of as the collective (distributed) wisdom of an entire subsystem concerning the routing-relevant connectivity among the components of that subsystem.)
- A routing algorithm that uses the information contained in the routing information base to derive routes between end systems

End systems (ESs) and intermediate systems (ISs) use routing protocols to distribute ("advertise") some or all of the information stored in their locally maintained routing information base. ESs and ISs send and receive these routing *updates,* and use the information that they contain (and information that may be available from the local environment, such as information entered manually by an operator) to modify their routing information base.

The routing information base consists of a table of entries that identify a *destination* (e.g., a network service access point address); the subnetwork over which packets should be forwarded to reach that destination (also known as the *next hop,* or "next hop subnetwork point of attachment address"); and some form of *routing metric,* which expresses one or more of the characteristics of the route (its delay properties, for example, or its expected error rate) in terms that can be used to evaluate the suitability of this route, compared to another route with different properties, for conveying a particular packet of class of packets. The routing information base may contain information about more than one "next hop" to the same destination if it is important to be able to send packets over different paths depending on the way in which the "quality of service" specified in the packet's header corresponds to different values of the routing metric(s).

The routing algorithm uses the information contained in the routing information base to compute the actual routes ("next hops"); these are collectively referred to as the *forwarding information base.* It is important to recognize that the routing information base is involved in computations that take place in the "background," independent of the data traffic flowing between sources and destinations at any given moment; but the forwarding information base is involved in the real-time selection of an outgoing link for every packet that arrives on an incoming link and must therefore be implemented in such a way that it does not become a performance-killing bottleneck in a real-world intermediate system (router).

Figure 1 illustrates the decomposition of the OSI routing function as it is represented in ISO/IEC TR 9575, the OSI Routing Framework.

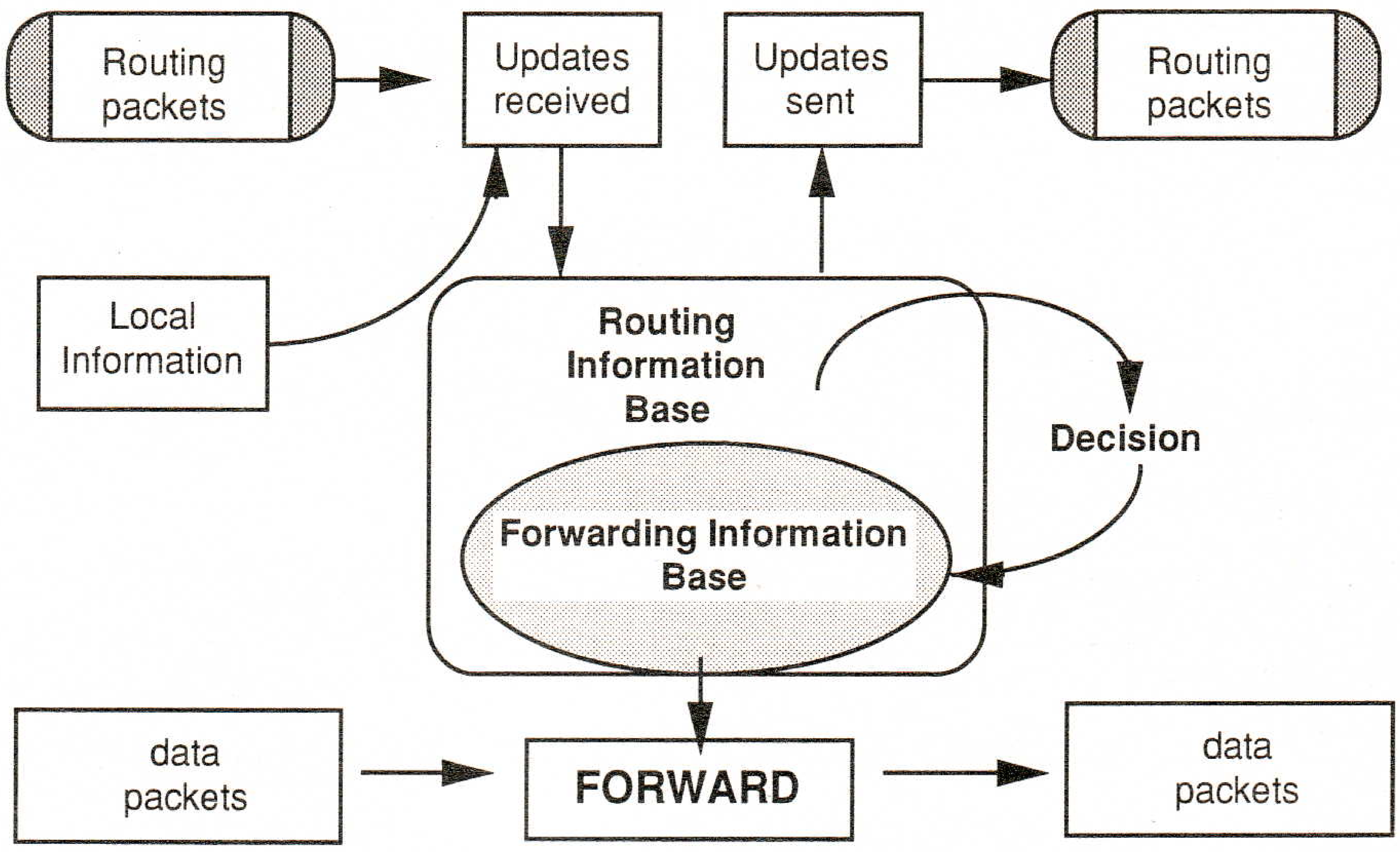


Figure 1: Decomposition of the OSI Routing Function

Three tiers

No system—certainly not an end system, which is supposed to be devoted primarily to tasks other than routing—can maintain a routing information base containing all the information necessary to specify routes from any "here" to any "there" in the entire global Internet. Neither is it possible to design a single routing protocol that operates well both in local environments (in which it is important to account quickly for changes in the local network topology) and in wide area environments (in which it is important to limit the percentage of network bandwidth that is consumed by "overhead" traffic such as routing updates).

continued on next page

Introduction to Routing *(continued)*

The OSI routing architecture is consequently hierarchical, and is divided into three functional tiers:

1 *End-system to intermediate-system routing* (host-to-router), in which the principal routing functions are discovery and re-direction.

2 *Intradomain intermediate-system to intermediate-system routing* (router-to-router), in which "best" routes between ESs within a single administrative domain are computed. A single routing algorithm is used by all ISs within a domain.

3 *Interdomain intermediate-system to intermediate-system routing* (router-to-router), in which routes are computed between administrative domains.

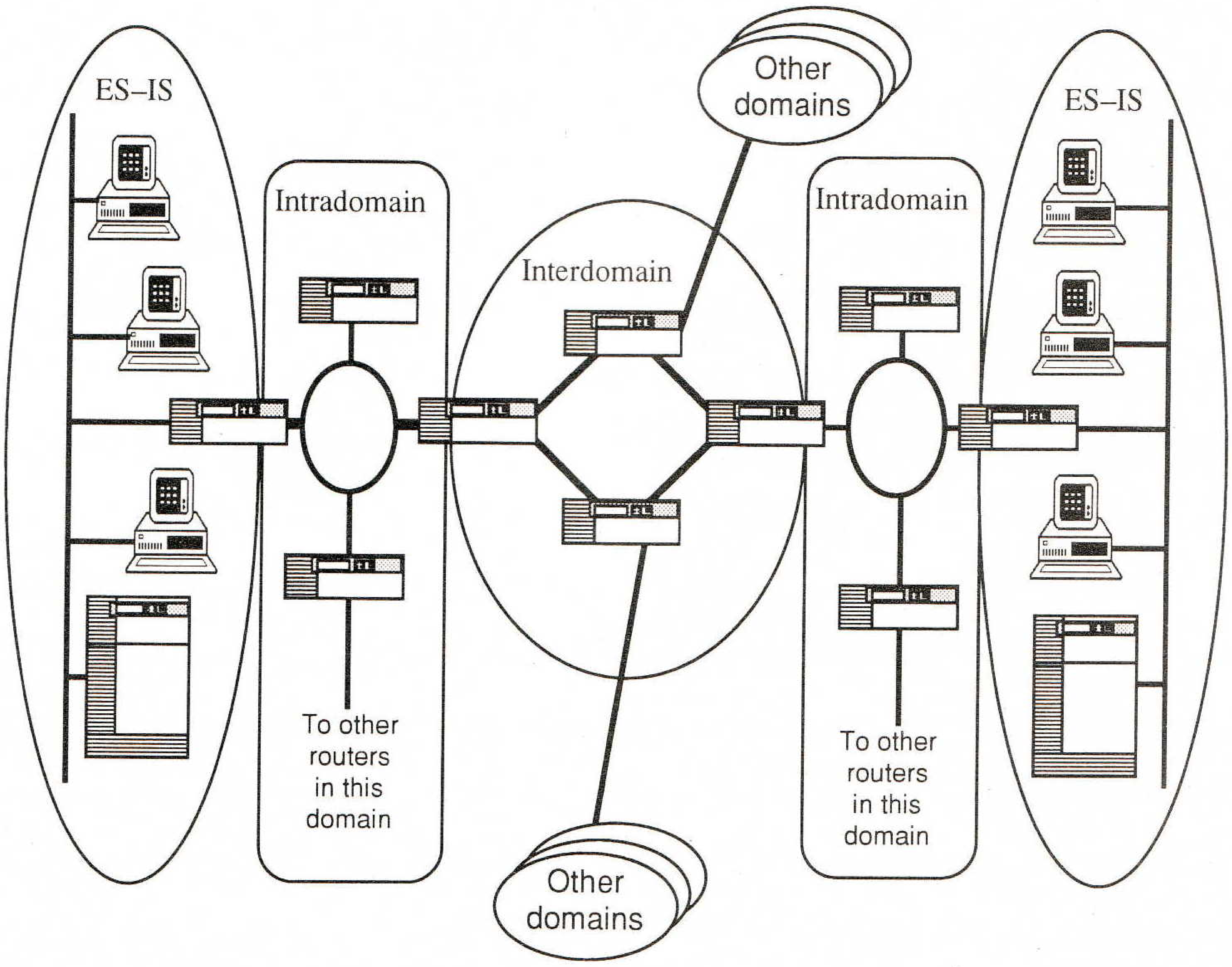


Figure 2: OSI Routing Architecture

Mechanism

In Figure 2, end systems discover and communicate with the intermediate systems to which they are directly connected (by dedicated or dial-up point-to-point links or by multiaccess local or metropolitan area networks) in the outermost level of the hierarchy; intermediate systems communicate with other intermediate systems within a single routing domain in the levels of the hierarchy next closest to the center; and in the center, intermediate systems communicate with other intermediate systems across routing domain boundaries.

The decomposition illustrated in Figure 2 is not arbitrary. At each level of the hierarchy, a different set of imperatives governs the choices that are available for routing algorithms and protocols. In the OSI routing architecture, end systems are not involved in the distribution of routing information and the computation of routes, and hence, the participation of end systems in routing is limited to asking and answering the question "Who's on this subnetwork with me?" (On broadcast subnetworks such as most local area networks, this inquiry typically begins with the more or less Cartesian assertion "I broadcast [multicast], therefore I am . . .")

Within a single routing domain, the hegemony of a single administration (and a correspondingly consistent set of routing policies) argues in favor of using a single routing protocol, i.e., one which is based on a link-state algorithm, which provides every intermediate system with complete knowledge of the topology of the routing domain. Between routing domains that may be controlled by different (possibly even antagonistic) administrations, issues of security (including control over the extent to which information about the topology of one domain is propagated to other domains) outweigh most others, and the argument in favor of distributing complete topology information to all intermediate systems, so compelling when selecting an intradomain routing protocol, misfires for the very reason that concealing or withholding information is often as important as distributing it. It is important to recognize that the analysis that leads to one conclusion in the intradomain context does not necessarily hold when it is transplanted to the interdomain context.

Which is better?

Fortunately, with respect to routing, it has not been difficult to refute the simplistic argument that if link-state routing is the best choice for the intradomain level it must be the best choice for the interdomain level. Ten years ago, however, a similarly simplistic argument destroyed the opportunity for OSI to standardize one of the best features of the TCP/IP internetwork architecture—the combination of a connectionless (datagram) internetwork protocol (which could be operated efficiently over any underlying network technology, whether based on datagrams or virtual circuits) with a connection-oriented end-to-end transport protocol. The OSI position at that time was that a connection-oriented service at the transport layer "naturally" mapped to a connection-oriented service at the network layer, as if this were something inherent in the very architecture of a layered model. The OSI community wasted years dealing with this red herring, which was intended to divert attention from the fact that a large segment of the OSI community believed that the service provided by the network layer was an end-to-end transport service. The TCP/IP community, unencumbered by such nonsense, happily expanded to fill the resulting vacuum.

ES–IS routing

ES–IS routing establishes connectivity and reachability among ESs and ISs attached to the same (single) subnetwork. Limiting the scope of routing in this manner allows an ES to play a simple role in the overall routing process and leaves most of the ES's resources available to support end-user applications (which is, presumably, the *raison d'être* of an ES). ESs are commonly attached to *multiaccess subnetworks,* such as IEEE 802 local area networks (LANs) and metropolitan area networks (MANs), creating topologies that are both highly connected and densely populated (with ESs); the protocols and algorithms that are appropriate for routing in this environment are very different from those that are appropriate for routing in the wide area environments served by intermediate system to intermediate system routing.

At this level of routing, the two critical (closely related concerns are discovery (who is out there?) and reachability (with whom is it possible to communicate?) Within a single subnetwork, an ES is one "hop" away from any ES or IS connected to the same subnetwork, so the only information an ES needs in order to reach either destination ESs on the same subnetwork or ISs that will forward packets to destination ESs on other subnetworks is the "hardware interface" or *subnetwork point of attachment* (SNPA) addresses of the ESs and ISs attached to the subnetwork.

continued on next page

Introduction to Routing *(continued)*

Intradomain IS–IS routing

Intradomain IS–IS routing establishes connectivity among intermediate systems within a single authority, the *administrative domain*. An administrative domain is composed of one or more *routing domains*. Each routing domain consists of a set of ISs and ESs; ISs within a routing domain use the same routing protocol, routing algorithm, and routing metrics.

At this level of routing, the critical concern is the selection and maintenance of best paths among systems within the administrative domain. ISs are concerned about route optimization with respect to a variety of metrics and about the trade-off between (1) the cost of distributing and maintaining routing information (which increases as the granularity of the information approaches "a separate route for every source/destination pair for every value of the routing metric[s]") and (2) the cost of actually sending data over a particular route (which increases if the available routing information causes data to be sent over a "suboptimal" route).

Interdomain IS–IS routing

Interdomain IS–IS routing establishes communication among different administrative domains, enabling them to control the exchange of information "across borders." In most circumstances, it is common to think of routing as something that tries to make it "as easy as possible" for two systems to communicate, regardless of what may lie between them. Interdomain routing, on the other hand, plays the paradoxical role of facilitating communication among open systems for which communication is a (politically) sensitive activity, involving issues of cost, accountability, transit authorization, and security that can produce highly counterintuitive answers to what look like simple technical questions. (Within the purview of a single network administration, it is considered to be a very good and useful thing for the network to automatically reconfigure itself to route traffic around a failed link onto an alternate path. In an interdomain configuration involving mutually suspicious network administrations, however, it may be the worst possible thing for the network to switch traffic to an alternate path "automatically" without first clearing the change with the legal departments of both parties. This conundrum has led one of the authors to claim that the only large-scale interdomain routing protocol that is likely to be deployed in the near future will be implemented as an army of lawyers on bicycles.)

At this level of routing the critical concern is the maintenance and enforcement of policies that govern, for example, the willingness of an administrative domain to (1) act as a transit domain for traffic originating from and destined for other administrative domains, (2) receive information from sources outside the administrative domain and deliver them to destinations within the administrative domain, and (3) forward information from within the administrative domain to destinations outside the administrative domain. Policies concerning 1, 2, and 3 can be derived on the basis of cost, access control, and regulatory concerns. The hierarchical relationship of the OSI routing protocols is depicted in Figure 3.

Within the OSI routing framework, it is possible for different routing domains within a single administrative domain to run different intradomain routing protocols, and it is also possible to operate different ES-IS protocols within different areas of the same routing domain. At present, however, OSI defines only one standard routing protocol for each of the three levels of the hierarchy.

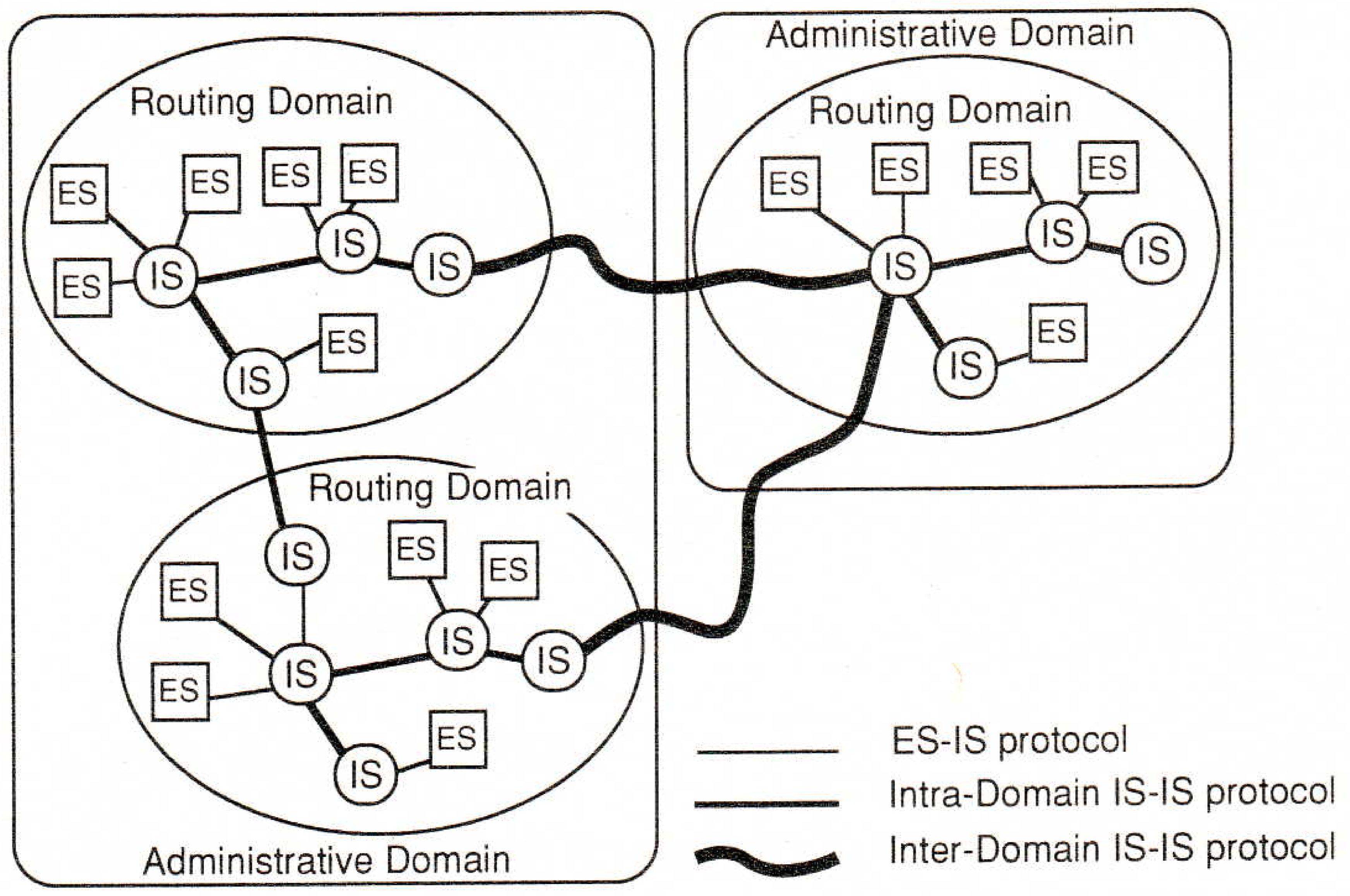


Figure 3: Hierarchical Relationship of OSI Routing Protocols

TCP/IP Routing Architecture

Through a process of evolution—in which some of the ideas that led to features of the OSI routing architecture originated with TCP/IP, developed into OSI standards, and returned to be adopted by the TCP/IP community—the TCP/IP routing architecture today is almost identical to the OSI architecture. The TCP/IP world began around a single network (which didn't require much in the way of routing), and grew into the "core"-based ARPANET, with individual networks connected to a single backbone (composed of "core gateways") as "stubs." Multiple organizations began offering IP transit services, and for a time, it was difficult to tell whether the Internet was a "mesh" of backbone networks or a hierarchy. A three-tier hierarchy was gradually introduced as the NSFNET grew and supplanted the ARPANET: the NSFNET served as a national backbone, and midlevel networks or "regionals" provided transit services to and from the IP networks whose directly connected hosts served as the sources and sinks of Internet traffic.

Today, the TCP/IP routing architecture looks very much like the OSI routing architecture. Hosts use a discovery protocol to obtain the identification of gateways and other hosts attached to the same network (subnetwork). Gateways within *autonomous systems* (routing domains) operate an *interior gateway protocol* (intradomain routing protocol), and between autonomous systems, they operate *exterior* or *border gateway protocols* (interdomain routing protocols. The details are different but the principles are the same.

Learn more in tutorial T6: "Open Systems Networking: TCP/IP and OSI," on Monday and Tuesday.

DAVID M. PISCITELLO works on broadband data services—Frame Relay, SMDS, Cell Relay/ATM—and network management at Bellcore. Involved in OSI-TCP/IP development since 1979, he participated in OSI/Internet standards development and has written several OSI and Internet standards including ISO CLNP and RFC 1209. A member of the IESG, he is the Internet area director for the IETF.

A. LYMAN CHAPIN graduated from Cornell University in 1973 with a B.A. in Mathematics, and spent the next two years writing COBOL applications for Systems & Programs (NZ) Ltd. in Lower Hutt, New Zealand. He joined the newly-formed Networking group at Data General Corporation in 1977. He moved to Bolt Beranek and Newman in 1990 as the Chief Network Architect in BBN's Communications Division. He is the chairman of ANSI-accredited task group X3S3.3, responsible for Network and Transport layer standards, since 1982; chairman of ACM SIGCOMM since July of 1991; and area director for Internet standards and procedures of the IESG.

Building Gigabit Network Interfaces

by Craig Partridge, BBN Systems and Technologies

Introduction

It is widely accepted that gigabit networks will have an effect on the operating systems and architectures of the computers, or *hosts,* that are attached them. The combination of higher bandwidth networks and increased application requirements for both bandwidth and performance guarantees suggests that we will expect more from our computers.

This article studies the particular problem of building hardware interfaces so that data can be moved between operating system and the network at gigabit speeds.

The Model Machine

One common mistake when studying the problem of interfacing hosts to networks is to assume networks are getting faster while hosts stay the same as today's systems. This kind of thinking typically leads to disastrous visions of plugging a gigabit network into a 20Mhz personal computer and desperate calls for special hardware to mediate between the massive network bandwidth and the poor overworked CPU.

In reality, gigabit networks will be (and are) attached to much more powerful machines. By 1996, workstation class processors are expected have instruction cycle times of about 1 nanosecond, and be capable (at peak rates) of performing one to two billion instructions per second (BIPS). (To make their terminology parallel with *giga*bits, some researchers have taken to referring to GIPS, for "*giga* instructions per second." This term is a linguistic barbarism (a misuse of the prefix *giga-*), and inconsistent with current terminology (which counts instructions in thousands, millions, billions and trillions. See, for example, the DEC Alpha processor, [1] whose designers are quite clear about their goal of achieving such high clock rates.)

Amdahl's rule of thumb states that each instruction per second requires a byte of memory and a bit of IO. Given general trends in memory costs, it is not unreasonable to expect these 1 BIPS processors to have close to 1 gigabyte of main memory. And developments in fast file systems, parallel disk systems such as *Redundant Arrays of Inexpensive Disks* (RAID), and file caching, make it clear that building file systems that can move data at gigabit rates is feasible. So the puzzle for this chapter is not how to make all the parts of the host fast—many will be fast already. Rather the problem is how to make sure that networking is fast too. The next several sections look at key pieces of the system and how they might be optimized.

The costs of moving data

In its simplest form, all networking does is take data from an application and send it over a network, or take data from a network and give it to an application. As a result, a large part of the cost of networking is the cost of moving data around between the network interfaces, the host operating system, and the networked applications.

Unfortunately, as a number of studies have shown, the relative cost of moving data, and in particular, copying data is increasing. For example, Ousterhout examined operating system performance on RISC processors and found that data copies were far slower than the processor's rated performance would lead one to suspect. [2]

The reason that data copies are becoming expensive is that memory speeds have not kept pace with improvements in processor performance (see, for example, the discussion on pp. 426–7 of [3].

Because memory speeds have not kept pace, it is currently the case that a processor can execute as many as four instructions in the time it takes to read one word from main memory.

Newer processors contain a number of features that try to hide this disparity in speeds. One feature is *super-scalar* architectures, which permit multiple independent instructions to execute in the same clock cycle. Thus if an instruction is loading data from memory and the next instruction is doing something unrelated to the instruction loading memory, both instructions can execute in parallel. Another feature is *scoreboarding*. In a scoreboarded architecture, instructions that load data into a register do not wait for the data to come back from memory, but rather mark the register as awaiting data. The processor then continues to execute new instructions. The register gets filled with data when the memory access completes. Only if an instruction accesses the register before the memory access has completed does the processor stop execution (or *stall*). Both super-scalar architectures and scoreboarding are of primary benefit when reading data. For writing data, processors typically cache writes to memory, so that unless there are a lot of writes at once, the processor doesn't have to wait for the write to complete.

Each of these features, however, is of limited help if the task is copying a lot of data (e.g., a buffer of application data) from one place in memory (such as the application's memory) to another (e.g., networking buffers). When copying large amounts of data, the processor's ability to copy is limited not by how fast it can issue read and write instructions, but by the peak speed at which the system memory can serve those instructions. An important feature of any host's operating system is that it be designed to minimize the number of times that large amounts of memory have to be copied, because these copies are limited by the system's memory speed, rather than processor speed.

Reducing memory copy costs

Given that memory copies cost a lot, how can memory copies be minimized? The fundamental problem is to find a way to be able to copy data directly from the applications' buffers into the interfaces that transmit the data onto the network. We'll look at two ways to achieve this goal: memory mapped buffers on the interface, and copy-on-write memory management combined with DMA.

The Afterburner interface

Suppose that one implemented a network interface so that it looked like part of a processor's main memory, and the interface's memory was used as kernel buffer space. Then the two-copy process of copying data from application memory to kernel buffers, and then from kernel buffers to interface memory, could be implemented as a single copy from application memory to interface memory. Similarly, for inbound packets, data could be copied straight from interface memory to application memory. The number of memory copies would be halved, and since memory copying is relatively slow, performance would presumably improve. (In fact, in test cases, performance has been shown to nearly double). The performance advantages of such an interface were pointed out by Van Jacobson in his proposal for a WITLESS interface. [4]

WITLESS

(WITLESS stands for a *W*orkstation *I*nterface *T*hat's *L*ow-cost, *E*fficient, *S*calable and *S*tupid). Jacobson had studied protocol implementations and concluded that a large amount of the time spent sending and receiving packets was lost in the device drivers that took a large amount of time to control the interface (e.g., start transmissions and receive new packets), and moving data from the operating system to the interface hardware.

continued on next page

Gigabit Network Interfaces *(continued)*

His WITLESS design was designed to minimize both the memory copying and control costs. Based in large part on Jacobson's ideas, Banks and Prudence of Hewlett-Packard implemented an experimental FDDI interface they called *Medusa,* and then a general interface design called *Afterburner.* [5], [6] Both interfaces are for HP 700 workstations.

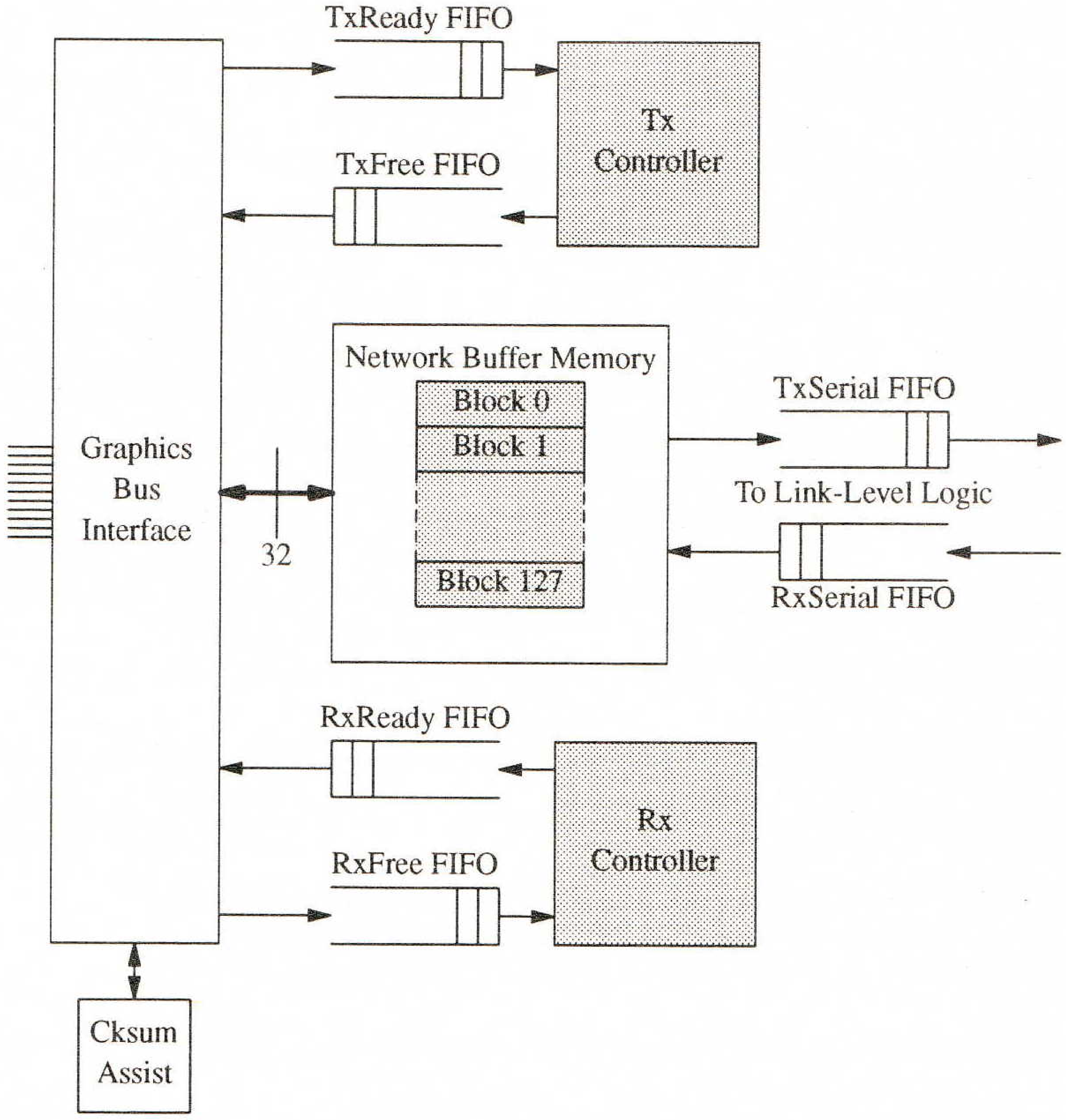


Figure 1: The Afterburner Interface design

Organization

The Afterburner design is illustrated in Figure 1. On the left side, the board is plugged into a memory bus with a 32-bit data path. (The Afterburner is actually plugged into the graphics bus and maps its memory into the processor's I/O memory space, both to avoid having to support error correction, which was required of memory interfaces in the HP 9000 series 700 workstation, and to avoid difficult interactions with the processor's memory cache). At the center of the interface is a triple-ported video memory, with two serial ports and one parallel port. The parallel port is used to support fast access to the graphics bus, and the two serial ports provide transmit and receive paths to the link-level chipset (not shown) off the right side of the interface. (The link-level chipset contains the network-specific hardware to put data onto the network, such as ATM over SONET or FDDI encoding hardware). Memory is organized in fixed-size 8 KB buffers. Fixed-size buffers have the dual advantage of making memory management simple (no fragmentation concerns) and making it possible to specify a buffers length and address in a single word.

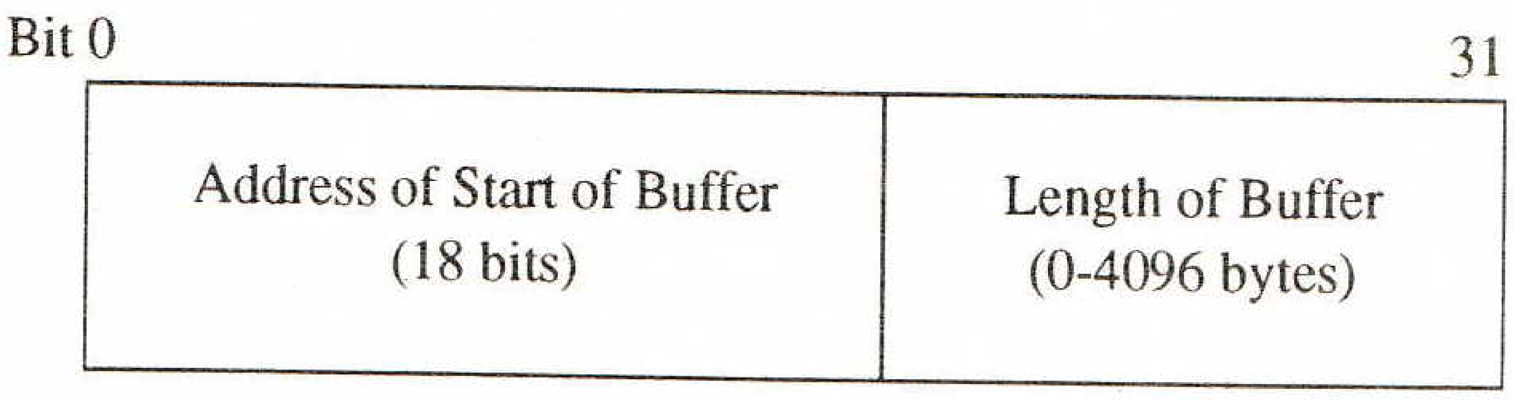


Figure 2: Afterburner FIFO Control Word format

To transmit a packet using the Afterburner, a host can read a word from free transmission buffer (*RxFree*) FIFO, which contains the address of a free buffer. The host fills the buffer with the packet header and the application data, and then writes a one word message containing the high-bits of the address combined with the length of the buffer in the low order bits (as shown in Figure 2) into the transmission ready (*RxReady*) FIFO. The Transmission controller takes packets from the ready FIFO and streams their data onto the network out the transmission serial port of the buffer memory. When the buffer has been sent, it is freed by placing it back in the *RxFree* FIFO. Receiving works similarly. Inbound packets are placed in the receive ready (*RxReady*) FIFO. The processor is only interrupted when the FIFO goes from empty to non-empty (multiple packets should be serviced in the same interrupt due to context switching costs). When the host is done with the buffer, it writes the buffer address into the receive free (*RxFree*) FIFO.

Observe that because all control communications take the form of atomic single-word writes and reads, there's no need for ready semaphores or other complex driver support.

Experiments with the Medusa interface (which is essentially an Afterburner for FDDI) have shown two impressive results. First, by comparing the throughput using the Medusa interface using the old-style BSD *mbuf* code, and using revised code that uses the Medusa's memory as the source of kernel buffers, it was shown that eliminating one memory copy effectively doubled throughput. Second, a relatively slow workstation (1990 vintage) with the Medusa interface was capable of transmitting at full FDDI bandwidth with processing time left over. Achieving full FDDI bandwidth had previously been thought to require a far faster workstation. Experiments using two Afterburner boards cabled together (with no link-level chipset) have achieved nearly 200 Mbps throughput.

Copy-on-write with DMA

The Afterburner/WITLESS approach to data copying is often referred to as *programmed IO* since the IO data copies are done by the processor. Another way to design computer systems is to make it possible for data copies to be done between interfaces without disturbing the processor, using a technique known as *Direct Memory Access* (DMA). For these systems, several implementors have had success at minimizing data copies using a combination of a virtual memory technique called *copy-on-write* (COW) to manage the data buffers, and a DMA network interface.

Copy-on-write is a technique to permit pages in a virtual memory system to be shared inexpensively. If two or more applications, or an application and the operating system both want access to the same page of data, they are both given pointers to the same page, and the page is marked copy-on-write. If a user of the page tries to change (or write to) the page, then the page is copied and the user is given a private copy, so that the user's changes do not affect the data of the other users of the page. Since most sharing is done to allow multiple access for the purposes of reading (for example, multiple instances of the same program sharing the same instructions and static data structures) COW can generally reduce the amount of copying required in a virtual memory system. [7]

DMA is a data transfer technique in which interfaces on a bus can transfer data to and from the processor's memory without disturbing the processor. DMA schemes are typically optimized for moving bursts of data (e.g., 32 words of data) at a time.

continued on next page

Gigabit Network Interfaces *(continued)*

DMA is typically a mixed blessing, because while it makes it possible to copy memory without disturbing the processor, it sometimes limits processor access to main memory during the data burst, and often requires complex setup code be executed to do a transfer.

However, despite some of the limitations of DMA, high-performance implementations have been achieved by combining a copy-on-write with DMA. The essential idea is that when an application asks for data to be transmitted, the operating system marks the pages containing the application's data as copy-on-write, and uses those pages as the operating system's data buffer. So instead of copying data from application space to operating system space, the data is shared. The operating system generates the packet headers in a bit of operating system memory, and then instructs the interface to DMA the headers and data from the shared pages into a packet buffer and send them. Assuming the application does not write to the pages containing the data it just sent, and force the pages to be copied, this scheme, like programmed IO, requires just one memory copy—the DMA of the data into the interface. (If the application does not cooperate and writes on the page before the data has been sent, then performance is typically rather poor.)

On the inbound side, the operating system keeps the interface informed about what pages are free. When a packet comes in, the interface DMAs the packet data into a free page and then notifies the operating system that a packet has arrived. The operating system removes the packet headers, and if the application's receiving buffer is page-aligned, simply changes the application's page table to replace the receiving buffer's page with the page containing the packet.

Observe that this scheme requires that the application take care to only use buffers that are page-aligned and an integral number of pages long, so some application cooperation is required. In some cases, however, language compilers and loaders can be modified to recognize data buffers and align them properly without changing the application.

Like the Afterburner approach, this approach to interface design has yielded high performance.

Eliminating all data copies?

Both the Afterburner/WITLESS approach and the COW/DMA approach reduce the number of required data copies for inbound or outbound packets to one copy between application space and interface memory, either using programmed IO or DMA.

It is worth noting that researchers have considered trying to eliminate the data copy altogether. The basic idea is to combine the page management technique of the COW/DMA approach with the on-board memory of the Afterburner/WITLESS approach such that the interface memory can be used as regular pages in application memory space. The application would pre-allocate some interface pages to use as its sending data buffers. Thus by placing data in its buffers, the application is automatically placing data in the interface memory, ready to be transmitted. Similarly inbound packets would not have to be DMA'ed, they would already be in pages that could be remapped into the receiving application's space.

This approach is feasible, but has one major drawback. If interface memory is of limited size, then it may have to be managed by the applications that use it. In particular, applications would have to carefully manage the network buffers they use, with calls to the operating system to allocate and free buffers as needed. Many system designers feel that part of the goal of an operating system is to conceal these sorts of hardware-specific issues from applications.

References

[1] R. L. Sites, "Alpha Architecture Reference Manual," Digital Press, 1992.

[2] John K. Ousterhout, "Why Aren't Operating Systems Getting Faster As Fast as Hardware?" Proceedings of 1990 Summer USENIX Conference, Anaheim, California, USA, June 11–15, 1990.

[3] J. L. Hennessy, D. A. Patterson, "Computer Architecture: A Quantitative Approach," Morgan Kaufmann, 1990.

[4] V. Jacobson, "Tutorial Notes from SIGCOMM '90," Philadelphia, USA, September 1990.

[5] D. Banks, M. Prudence, "A High Performance Network Architecture for a PA–RISC Workstation," *IEEE Journal on Selected Areas in Communications,* February 1993, Volume 10, No. 1, pp 191–202.

[6] G. Watson, D. Banks, C. Calamvokis, C. Dalton, A. Edwards, J. Lumley, "Afterburner: Architectural support for high performance protocols," *IEEE Network Magazine,* Volume 7, No. 4.

[7] J. M. Smith, G. Q. Maguire, Jr.," Measured Response Times for Page-Sized Fetches on a Network," *ACM SIGARCH Computer Architecture News,* Volume 17, No. 5, September 1989, pp 71–77.

CRAIG PARTRIDGE holds an A.B., M.Sc. and a Ph.D from Harvard University. Since 1983 he has worked for Bolt Beranek and Newman on a variety of networking related projects including CSNET, the NSF Network Service Center (NNSC), and various projects concerned with distributed systems, IP transport protocols, and gigabit networking. He is a member of the Internet End-To-End Research Group, the Internet Engineering Task Force and a Senior Member of the IEEE. He is the past editor of ACM *Computer Communication Review* and the current editor of IEEE *Network Magazine.* E-mail: `craig@aland.bbn.com`

To learn more, attend tutorial T19: "Gigabit Network Architectures," given by Craig Partridge on Monday and Tuesday, August 23 and 24.

Windows Sockets: Where necessity is the mother of *re*-invention

by Martin Hall, JSB Corporation

Introduction

The computer software industry has a bad habit of responding to necessity without paying sufficient attention to existing solutions. In 1991, it was becoming increasingly apparent that TCP/IP was *the* solution for inter networking heterogeneous platforms. Whilst open systems interoperability was providing increased connectivity to the marketplace, similar openness in the application development environment on PCs was not present. Vendors of TCP/IP implementations on DOS-based PCs had overlooked their own "backyard," Every vendor of TCP/IP for a PC had an API for application developers but all these APIs were different. For application developers to have to do special work to support each different TCP/IP implementation was obviously counter productive.

Since software development on PCs is done increasingly within the Microsoft Windows environment, it seemed sensible to use facilities within Microsoft Windows to put an end to this proliferation of different APIs. Additionally, it was important not to invent a totally new API but to seize on the existing knowledge and experience of Berkeley Sockets as a network API.

This situation led to the *Windows Sockets* specification. Windows Sockets defines a standard, open network programming interface for Microsoft Windows and Windows NT which is based on the "socket" paradigm popularized in the *Berkeley Software Distribution* (BSD) from the University of California at Berkeley. It includes both familiar Berkeley socket style routines and a set of Windows-specific extensions for further integration with the message-driven Windows and Windows NT environments.

Windows Sockets emerged from a cooperative effort involving competing PC TCP/IP vendors and application developers. 12 months of meetings and considerable Internet discussion resulted in the Windows Sockets Version 1.1 specification which was released in January 1993. For details of how to get involved in the Windows Sockets group and/or how to retrieve the specification and associated documentation and software, see the end of this article.

Technical overview

The Windows Sockets specification defines a set of APIs for implementation by multiple vendors. Typically, vendors of network transport implementations such as TCP/IP will provide an instance of this API set. Each Windows Sockets implementation is incorporated in a Windows *Dynamic Link Library* (DLL). There are thus multiple Windows Sockets DLLs being created, one from each network transport vendor.

Architecture

Figure 1 helps illustrate where the Windows Sockets DLL sits relative to other software components. As you can see from this diagram, the method by which the Windows Sockets DLL accesses a TCP/IP kernel is vendor specific.

Windows Sockets API categories

The APIs defined in Windows Sockets fall into 3 categories:

- Socket Routines
- Database Routines
- Microsoft Windows-specific Extensions

The Socket and Database APIs are very faithful to Berkeley Sockets 4.3 except where the architecture of Microsoft Windows has necessitated change. The Microsoft Windows-specific extensions are designed to allow application network communications to be event driven.

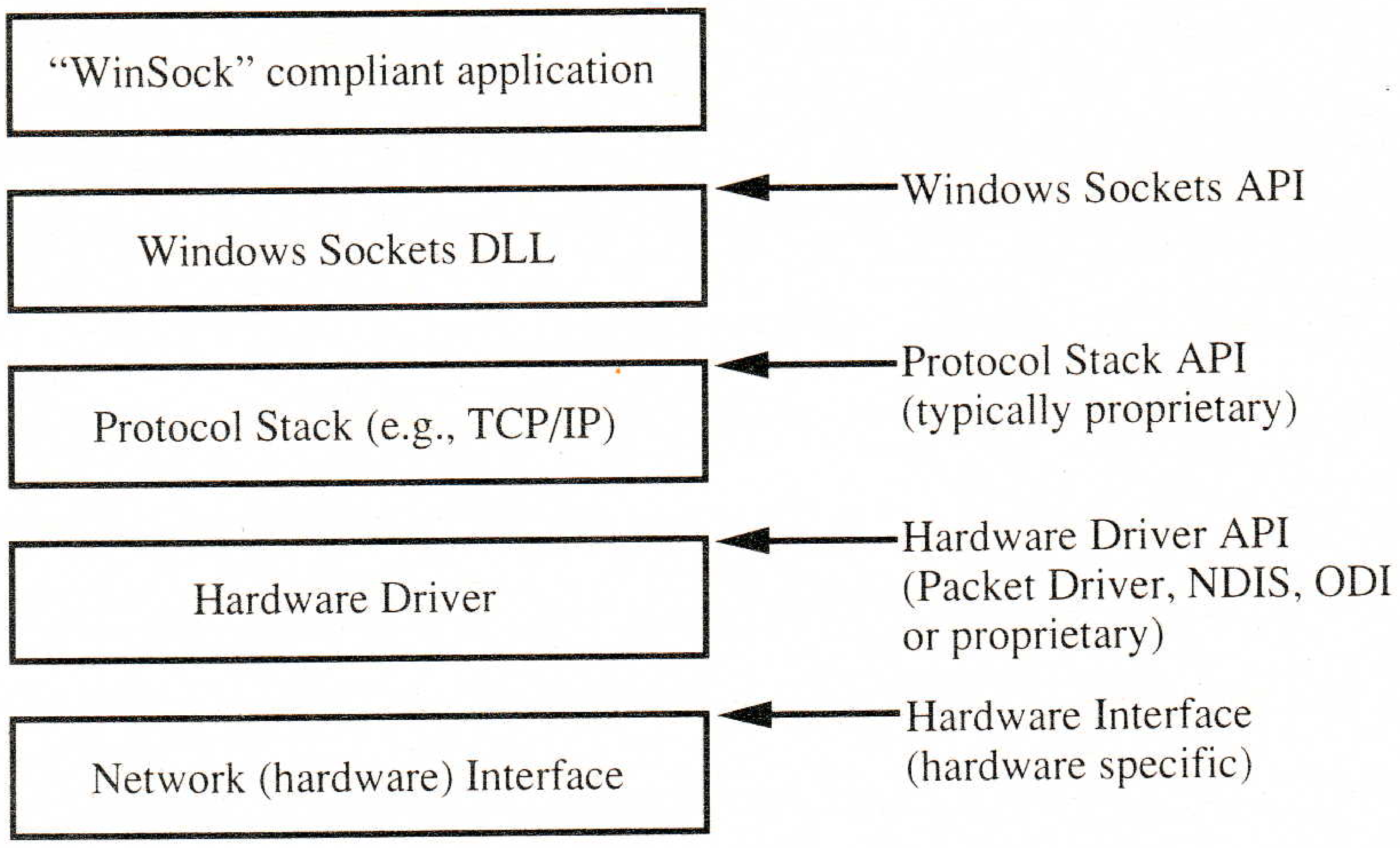


Figure 1: Window Sockets DLL in context

Socket routines

Windows Sockets implementations include support for the following Berkeley Socket APIs:

accept()
bind()
closesocket()
connect()
getpeername()
getsockname()
getsockopt()
htonl()
htons()
inet_addr()
inet_ntoa()
ioctlsocket()
listen()
ntohl()
ntohs()
recv()
recvfrom()
select()
send()
sendto()
setsockopt()
shutdown()
socket()

There are some subtle differences between socket operations in Windows Sockets and regular UNIX based BSD sockets. These can cause problems if you are not aware of them.

The names of 2 regular UNIX APIs which can operate on sockets, *close()* and *ioctl()*, had to be changed (to *closesocket()* and *ioctlsocket()*) since sockets are not necessarily file handles in Windows Sockets implementations.

Additionally, whereas sockets on UNIX are file descriptors with a signed value, sockets in Windows Sockets are unsigned integers.

continued on next page

Windows Sockets *(continued)*

This means that the following code is *dangerous* in Windows Sockets applications:

Bad code: `if ((s = socket()) < 0)`

Good code: `if ((s = socket()) == INVALID_SOCKET)`

The *select()* API in Windows Sockets operates on arrays of sockets rather than bit masks. This means you *must* use the *FD_???* macros for manipulation of *fd_set* structures. You risk major application problems and crashes if you don't.

One interesting manifestation of support for multi-threaded environments built into Windows Sockets is error code retrieval. In regular BSD UNIX sockets programming, error codes are retrieved via the external variable "errno." In Windows Sockets, because of complications that can arise in accessing such a variable in a multi-threaded environment such as Windows NT, error codes are retrieved instead via the API *WSAGetLastError()*. This is consistent with general error code retrieval in Windows NT and, as you might expect, has a sister API which allows you to explicitly set an error code, *WSASetLast - Error()*. You should therefore not expect the following code to work:

Bad code:

```
if ((n = recv(s, buf, 16)) < 0 && errno == EWOULDBLOCK)
```

Good code:

```
if ((n = recv(s, buf, 16)) < 0 && WSAGetLastError() ==
    EWOULDBLOCK)
```

One final important point to remember when writing a Windows Sockets application is that, under Microsoft Windows 3 at least, you are strongly advised to avoid using blocking sockets. The non-preemptive nature of Windows means that operations on blocking sockets under Windows 3 have to be implemented in the Windows Sockets DLL in such a way as to potentially relinquish the processor. If your application makes a Windows Sockets call on a blocking socket then it must be coded to cater for application reentrancy. This is obviously not a pretty scenario so you should stay away from blocking sockets. Under Windows NT, a true preemptive operating system, you can happily use blocking sockets without any such concerns.

Database routines

Windows Sockets implementations include support for the following "database" routines:

gethostbyaddr()
gethostbyname()
gethostname()
getprotobyname()
getprotobynumber()
getservbyname()
getservbyport()

How a particular Windows Sockets implementation resolves, for example, a host name is not mandated by Windows Sockets. The Windows Sockets implementation may use local host name resolution or DNS. This is, however, transparent to the application.

Your application must treat data areas returned by these functions as volatile and should thus copy data from these areas to local variables.

Microsoft Windows extensions

The following APIs include asynchronous implementations of existing Berkeley Sockets APIs together with other miscellaneous Windows Sockets specific APIs:

WSAAsyncGetHostByAddr()
WSAAsyncGetHostByName()
WSAAsyncGetProtoByName()
WSAAsyncGetProtoByNumber()
WSAAsyncGetServByName()
WSAAsyncGetServByPort()
WSAAsyncSelect()
WSACancelAsyncRequest()
WSACancelBlockingCall()
WSACleanup()
WSAGetLastError()
WSAIsBlocking()
WSASetBlockingHook()
WSASetLastError()
WSAStartup()
WSAUnhookBlockingHook()

Of primary importance amongst the above APIs are *WSAStartup()* and *WSACleanup()*. Your application must sign on to and sign off from a Windows Sockets DLL by using these APIs. *WSACleanup()* provides a straightforward signing off mechanism. *WSAStartup()* allows an application to do 2 things:

- Determine information specific to the particular Windows Sockets implementation it is using.
- Negotiate the version of Windows Sockets support it requires. This is a provision which will allow future applications that require functionality specific to a particular version to check whether they are running with a Windows Sockets implementation that supports such functionality.

Should you wish to maintain source code across both Windows and non Windows platforms, you would probably decide not to use any of the message-based functions (*WSAAsyncXXXX*).

However, the asynchronous APIs are extremely powerful and can result in elegant network aware Windows application code. To see how these APIs operate, we'll take a look at one asynchronous database function, *WSAAsyncGetHostByName()* and the central network event notification mechanism which is activated by *WSAAsyncSelect()*.

Data lookup

In common with the other asynchronous database lookup APIs, *WSAAsyncGetHostByName()* allows you to post interest in resolution of some data item (in this case a host name) and have the application notified by message of the completion of the lookup. *WSAAsyncGetHostByName()* is prototyped as follows:

```
HANDLE PASCAL FAR

WSAAsyncGetHostByName(HWND hWnd, unsigned int wMsg,
const char FAR *name, char FAR *buf, int buflen)
```

This API's return value represents an "asynchronous task handle." When the lookup is complete, the application's Window procedure identified by "hWnd" will receive the message "wMsg."

continued on next page

Windows Sockets *(continued)*

Standard parameters to the Window procedure contain the identifying "task handle" together with an error code if the lookup failed. If it was successful, then the buffer pointed to by "buf" will contain the result of the lookup. To further clarify the use of *WSAAsyncGetHostByName(),* consider the following code fragments.

```
Calling WSAAsyncGetHostByName()...

#define WM_GOTHOSTNAME          (WM_USER + 1)

HWND     hMyWnd;
HANDLE   hAsyncTaskID;
char     HostEntBuf[MAXGETHOSTSTRUCT];
if ((hAsyncTaskID =
     WSAAsyncGetHostByName(hMyWnd,
                           WM_GOTHOSTNAME, "RemoteHostName",
                           HostEntBuf, sizeof(HostEntBuf))) == -1)
{
    // Process Error
}

Receiving asynchronous notification of completion...

LONG FAR APIENTRY MyWndProc(HWND hWnd, UINT wMsg, UINT wParam,
                            LONG lParam)
{
    switch (wMsg)
    {
        case WM_GOTHOSTNAME:
            if (WSAGETASYNCERROR(lParam))
            {
                // Process Error
            }
            if (hAsyncTaskID = wParam) // Double check task handle
            {
                // Process data for "RemoteHostName"
                // now in HostEntBuf
            }
            break;
    }
}
```

Network events notification

WSAAsyncSelect() can be used as the central agent for determining network activity. The complete set of events on which your application can be notified are:

- Data is ready for reading
- Socket is ready for writing
- Out-of-band data has arrived
- An incoming connection has arrived
- An outgoing connection has completed
- An existing connection has been closed

Having created a socket for data exchange with a remote server, for example, an application may want to receive asynchronous notification of:

- The presence of incoming data sent by the remote server application.
- Connection closure.

To do this it would first post interest in receiving such notification by calling *WSAAsyncSelect()* something like:

```
#define WM_SOCKET_ACTIVITY        (WM_USER + 2)
HANDLE hMySocket;
HWND   hMyWnd;

    if (WSAAsyncSelect(hMySocket,
                       hMyWnd, WM_SOCKET_ACTIVITY,
                       FD_READ | FD_CLOSE) < 0)
    {
        // Process Error
    }
```

When incoming data arrives or the connection is closed by the remote server, the application's Window procedure would be called. The application's Window procedure would receive message *WM_SOCKET_ACTIVITY*. The arguments would identify a possible error code and the actual event which has occurred. The code would look something like:

```
LONG FAR APIENTRY MyWndProc(HWND hWnd, UINT wMsg, UINT wParam,
                            LONG lParam)
{
    switch (wMsg)
    {
        case WM_SOCKET_ACTIVITY:
            if (WSAGETSELECTERROR(lParam))
            {
                // Process Error
            }
            switch (WSAGETSELECTEVENT(lParam))
            {
                case FD_READ:   // Data ready for reading
                    recv(hMySocket, buf, buflen);
                    break;

                case FD_CLOSE:  // Connection has closed
                    close(hMySocket);
                    break;

            }
            break;
    }
}
```

WSAAsyncSelect() is an extremely powerful API. It is easy to use badly. You should code it in the simplest way you can for maximum clarity.

Windows Sockets components

The components which go to make up Windows Sockets are:

- *Development Components:*
  Windows Sockets specification—documents the APIs etc.
  WINSOCK.H—header file
  WINSOCK.LIB—import library

(The header and library files are provided by each supplier of a Windows Sockets DLL.)

- *Run time component:*
  WINSOCK.DLL—This is the name of the DLL provided by each Windows Sockets supplier.

continued on next page

Windows Sockets *(continued)*

Mailing list

A Windows Sockets electronic mailing list is maintained for technical discussions of the Windows Sockets specification and for announcements of the availability of new versions of Windows Sockets products, and of the specifications themselves. The list is maintained by Mark Towfiq at Microdyne. To subscribe, send e-mail to `winsock-request@microdyne.com`. To send mail to the list itself, e-mail to `winsock@microdyne.com`. The list is not moderated. It is suggested that you use signatures at the end of your messages, as some mailers do not decode the headers to distinguish the initial sender of a given message.

Newsgroup

The mailing list `winsock@microdyne.com` is gatewayed to the USENET newsgroup `alt.winsock`. This provides an alternative interface to those wishing to discuss Windows Sockets.

FTP sites

You can retrieve the Windows Sockets specification, Guide and various software from the following FTP sites via anonymous FTP:

```
microdyne.com          /pub/winsock
vax.ftp.com            /pub/winsock
SunSite.UNC.EDU        /pub/micro/pc-stuff/ms-windows/winsock
rhino.microsoft.com    winsock
```

For further reading

You may find the following Windows Sockets publications useful:

[1] "Windows Sockets: An Open Interface for Network Programming under Microsoft Windows," Martin Hall, Mark Towfiq, Geoff Arnold, David Treadwell, Henry Sanders. *The definitive specification, this is available from various sources including FTP sites detailed above.* Published January 20 1993.

[2] "A Guide to Windows Sockets," *A tour around Windows Sockets, this is available from various sources including FTP sites detailed above,* Martin Hall, June 1 1993.

[3] "Plugging into TCP/IP with Windows Sockets," Victor Volkman *Windows/DOS Developers Journal,* Vol. 3, No. 12, December 1992.

[4] "Untangling the Windows Sockets API," Mike Calbaum, Frank Porcaro, Mark Ruegsegger, Bruce Backman, *Dr. Dobbs Journal,* #197 February 1993.

[5] "The Windows Sockets API," Ralph Davis, Chapter 6 in *Windows Network Programming,* from "The Andrew Schulman Programming Series," Addison-Wesley Publishing Company, 1993.

[6] "Windows Sockets—Get Plugged in to Serious Network Programming," J. Allard, Keith Moore, David Treadwell, *Microsoft Systems Journal,* July 1993.

MARTIN HALL holds a BA and M.Sc in Computer Science. He is currently the Technical Vice President of JSB Corporation and chairs the Windows Sockets group. He can be reached as: `martinh@jsbus.com`

Learn more: Attend Session S56: "Windows Sockets" on Friday, August 27, at 10:30am.

A Map to Wireless Networking and Mobile Computing

by Rifaat A. Dayem, Altamont Research

Introduction

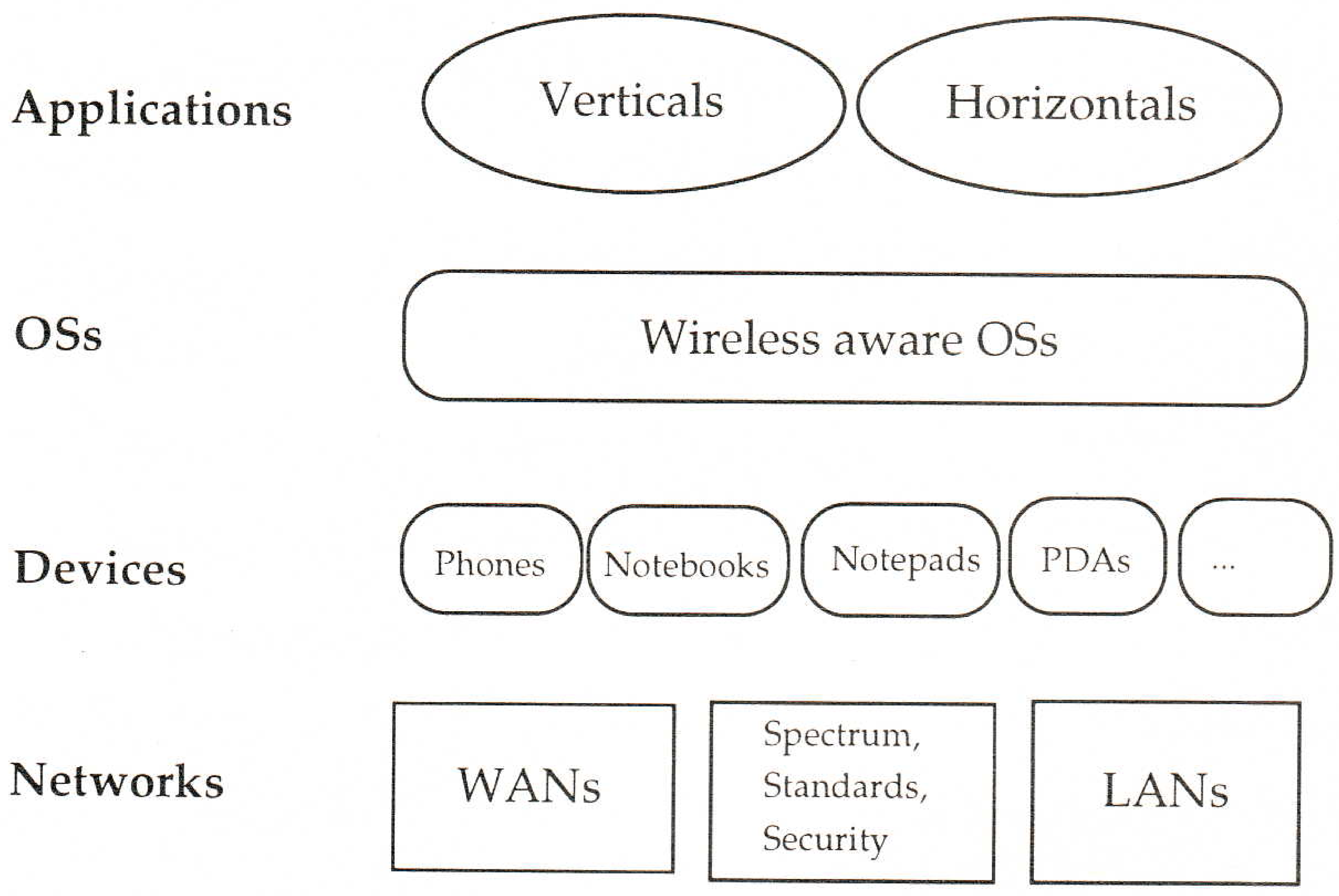
The fields of Wireless Networking and Mobile Computing are about the most exciting fields in our industry today. There are many companies and groups working on Wireless Networking and Mobile Computing. There are opportunities for large companies as well as numerous opportunities for small startups. This is what makes this field so interesting.

To bring the vision of Wireless Networking and Mobile Computing to life we need wireless wide area networks, and wireless local area networks. We need new spectrum for these networks. We need to develop standards including robust security standards for both WANs and LANS. We need operating systems that are aware of the wireless networks running underneath them. These OSs free the applications writers from the mystery of handling various weird and wonderful networking protocols and allow them to focus on developing powerful applications for both the vertical and horizontal markets.

Applications

Many people speak of "killer applications" that will make wireless take off. There may be no one or a few killer applications for a market such as this. This technology offers new ways of communicating that we are not familiar with and I would venture to guess that up to a third of the wireless applications we will be using 10 years from now are not ones we would think of today. That is the kind of technology wireless is. We have to experience it to see its full potential.

What is profitable today are vertical applications that solve specific problems such as managing fleets of delivery trucks and personnel, or serving a mobile field engineering team. The horizontal applications that all of us will want and not know how we ever got along without include e-mail and networked calendars and information on demand. For these horizontal applications to take off I believe we need a price point well below what is available today.



Wireless Taxonomy

Operating systems

Operating systems are the key to making writing applications for Wireless Networks and Mobile Computing a pleasure rather then a chore. Some of the key companies working in this area are Go, EO and General Magic.

continued on next page

Wireless Networking & Mobile Computing *(continued)*

Apple is developing their own operating system for the *Newton* product family. These operating systems can be applied to several platforms including pen computers. The pen computer interface is notebook-like with tabs on the side. Pointing to a tab goes to that section of the notebook, opens the right applications, and depending on what the user does may activate particular communications drivers, and queue up messages for sending. These messages may be electronic mail, a fax, or a file transfer on a LAN, or a database inquiry that may start with a local server and then look for the information elsewhere. How transparent these services are to the user is bound only by the imagination of the developer of the application. Can we finally free the user of the gobblygook of networking?

Platforms

Platforms for the end user are probably the most visible part of the industry right now. They include cellular phones, PCS (*Personal Communications Services*) phones, notebook computers, notepad computers, and the class of new devices known by many names including *Personal Digital Assistants,* (PDAs), *Information Appliances* (IAs), and *Personal Communicators* (PCs???). One thing is for sure about these devices, they are not short on what we have chosen to call them.

Looking closer at the names reveals the origins, culture, and philosophy behind the names as well as the likely shape that each device may take. I suspect that a Personal Digital Assistant from Apple will look quite different from a Personal Communicator from AT&T. I wonder how an Information Appliance from HP will evolve. One thing for certain, we will have a rich, diverse set of products to choose from as each sector of the industry invents its own version of the future. I, for one, am looking forward to participating in this exciting unveiling. It may not seem like much now, but I believe it will be an exciting evolution over this coming decade. Step aside Star Trek!

Back to earth, the networking fabric under all this is what will probably take the longest time to build. There are many companies already under full steam building wireless WANs, and wireless LANs.

Cellular networks

Wireless WANs are dominated by cellular networks. Cellular networks are dominated by voice traffic. Have you ever tried to send data over cellular today? It is a very iffy proposition at best. This is changing. There are cellular modems being developed that can convert the medium into a less hostile one for data. There are even chip sets now available for taming it.

Specialized Mobile Radio (SMR) operators such as ARDIS and RAM are providing wireless WAN data services today. Both these companies are hard at work expanding their networks and their base of applications. ARDIS is taking the approach of focusing on vertical applications today and saving horizontal applications for a later time. RAM is betting on horizontal applications right now, but is also hedging their bet by being open to vertical applications.

Nipping at their heals is CDPD, *Cellular Digital Packet Data.* This consortium is composed of the "who is who" in communications (6 out of the 7 RBOCs and McCaw Cellular) with one heavy weight computer manufacturers (IBM). Considering that IBM owns part of ARDIS, Bell South owns a lot of RAM, and AT&T just bought a chunk of McCaw Cellular, we see that this is a very friendly industry, or else everyone is hedging their bets!

Wireless LANs have their bevy of products. Most of them work in the ISM (Industry, Scientific and Medical) band right now until we do get the new User PCS band. NCR, Proxim, Windata, Motorola, and Infra-LAN are a few of the entrants in this field.

They all attend standards meetings so we can connect and roam in and out of networks provided by different suppliers. And many of them are involved in lobbying the FCC and the rest of the government to have enough bandwidth allocated for wireless networks to serve the applications we can envision today as well as the ones we have not thought of yet.

Spectrum issues

The FCC and the government will not be the bottle neck in this field. This technology is too hot and too important for the economy of this country to have the hands of the industry tied because there is no spectrum available. I would guess that the standards process will be the slowest leg. A lot of people are very hard at work to reach standards, it is just a huge task to reach agreement among so many differing groups.

The current spectrum activities include the new Personal Communications Services (PCS) spectrum allocation plan in the *Emerging Technologies* (ET) band. "User PCS" is receiving a special chunk of spectrum, 20Mhz to start. This spectrum will be for private networks. This means wireless LANs and wireless PBXs. Users of this band will not have to worry about getting a license, just type approval. This is great but has the potential for turning the band into a garbage band like citizen's band radio. To protect against this, WINForum, the lobby for User PCS has developed a spectrum etiquette to insure the continued usability of the band. The word "etiquette" implies politeness, and we know how far politeness will go if it is a question of staying on line or losing a connection. I have seen the etiquette, and it is a lion in sheep's clothing. It has teeth, and to get type-approved, a manufacturer will have to comply.

The one challenge we were not quite ready to meet this time around at WINForum was being able to share the spectrum between wireless LANs and wireless PBXs in an elegant fashion. The way it is now divides the spectrum into two equal parts. We did not reach this simplistic answer without a lot of hard work, but we were simply not smart enough yet to come up with a protocol that allows these two animals to live in the same forest yet. I guess we have to wait till the next time around to finally see the world of LANs and PBXs come together.

References

[1] Allen, R., "Wireless networking for the 1990s," *ConneXions,* Volume 4, No. 10, October 1990.

[2] Potts, J. J., "Moving Towards the Ultimate Telecommunications: Personal Communications Services (PCS)," *ConneXions,* Volume 6, No. 10, October 1992.

Dr. RIFAAT A. DAYEM is a pioneer in the field of wireless networking. He earned degrees in physics and engineering from Cornell and Stanford universities. At Bell Laboratories, he was instrumental in designing private network services. At Apple, he spearheaded the wireless networking effort. He is currently president of Altamont Research, a market analysis firm in Cupertino, California, specializing in wireless networking.

Learn more:

Learn more:	***Tutorial T22:***	***"Wireless Networking"***
	Workshop W1:	***"Mobile Communications"***
	Session S17:	***"Wireless Networking"***
	Session S26:	***"Nomadic Computing"***

CALS—Improving Enterprise Competitiveness

by Carolyn Wimple, Lawrence Livermore National Laboratory

Introduction

The Department of Defense (DoD) spends billions of dollars annually to store, maintain, and revise the technical data needed to support weapons systems. The vast majority of this data is on paper and is managed manually. *Computer-aided Acquisition and Logistic Support* (CALS) is a DoD and industry strategy to accelerate the transition from paper-intensive non-integrated product development design, manufacturing, and support processes to a highly automated, integrated mode of operation by developing (1) standards for data storage and exchange, and (2) automated systems to store, manage, and distribute this information to many and varied users across an enterprise.

Background

CALS was recommended by a joint DoD-Industry Task Force, and initiated by the Deputy Secretary of Defense in the mid-1980s. Originally focused on weapon systems and the generation, access, management, maintenance, distribution, and use of associated technical data, CALS is now recognized by many organizations and entities in both government and industry as beneficial to non-weapons related efforts. Technical data includes engineering drawings, product definition and logistic support analysis data, technical manuals, training materials, technical plans and reports, and operational feedback data. The scope of the CALS initiative includes support for activities such as contracting, work breakdown structures, design specifications, drawing preparation and release, testability, producibility, reliability, and maintainability, and other paper-producing processes.

The CALS strategy fosters the notion of creating data once and using it many times, through the use of a minimal set of broadly accepted national and international *platform independent* digital data standards. CALS facilitates data integration, exchange, and access among government and industry maintained databases, and eliminates the development of duplicate data. CALS provides the framework to integrate existing "islands of automation" within government and industry. CALS also provides the opportunity for reduced lead times and cost, higher quality potential, and a tool for achieving a competitive edge. CALS provides a method of revising and retrieving documents and other technical data through the several phases of a product's life cycle: initial product definition, demonstration and validation, engineering, production and deployment, and long-term operation and support. DoD is accelerating the implementation of these standards by requiring their specification in procurements of major new systems.

Characteristics

The three pervasive characteristics of the CALS Strategy are:

- Integration of process and data,
- Government and industrial networks providing on-line data access,
- Digital data delivery.

Integration is the primary characteristic. An integrated environment of information and technology streamlines processes and eliminates the duplication of redundant tasks. *On-line data access* through networks, the second characteristic, speeds up the acquisition process and facilitates the integration and coordination needed for producing quality products.

Digital data delivery is the third characteristic. Although on-line access to data will be the typical mechanism for information transfers of limited scope, bulk data transfers will still be necessary. This is why standards are critical to the advancement of CALS.

CALS objectives

The objectives of the CALS strategy are to improve the timeliness, reduce cost, and improve the quality of products and their supporting technical data. Achieving these objectives will lead to increased operational readiness and industrial competitiveness.

Timeliness

Reduced time to market and quick industry response can be achieved by development of integrated data, automation of plant facilities, and industrial networking. Digital bid packages, electronic bids and orders, electronic shipping documents and electronic funds transfers, all achievable via the *Electronic Data Interchange* (EDI) aspect of CALS, will reduce administrative lead times. Shared data environments, embodied in CALS concepts, enable shortened design, development, production and resupply times. Also, integrated planning, automated tool design and setup, and rapid parts support will reduce "out of service" times and increase product reliability.

Reduced costs

The high cost of sharing and re-entering paper-based graphical and textual information is reduced through the exchange and integration of CALS digital data. Paper will be replaced by accurate, timely, and cost effective digital technical information, which will be shared by multiple systems, with interoperability achieved through common system applications. The improved quality of these systems and data will also significantly lower the product life-cycle costs.

Improved quality

Improved product quality and data consistency will result from fewer errors in product design and manufacturing, achievable through the integration of key databases which support integrated product development functions in a near real time environment.

The CALS players

CALS receives strong support from many government agencies both nationally and internationally. The CALS *Industry Steering Group* (ISG) was established to serve as a single industrial focal point for the CALS/*Concurrent Engineering* (CE) activities. While the ISG operates as an adjunct to the *National Security Industrial Association* (NSIA), it is a multi-association effort comprised of over 1,000 volunteers representing over 200 companies and 18 professional associations and societies, all working to achieve a common goal of implementing CALS and CE. Membership in the ISG is expanding to other government agencies such as Energy, Transportation and NASA, that also procure complex engineered products. Over thirteen ISG regional interest groups operate throughout the US with the objective of bringing CALS to "the field" so that participation in the ISG can be expanded through representation and education. The Department of Commerce's Technology Administration sees CALS as the digital merging of the entire manufacturing process, covering all aspects of design, engineering, production, and support. CALS is also expanding internationally. In 1992, three CALS Expos were held in Australia, United Kingdom and France. Many countries have now set up CALS offices.

The *National Institute of Standards and Technology* (NIST) is working with the CALS program to develop and deploy standards that are vital for the exchange of digital information. Exchanged information must be as instantaneously understandable, interpretable and useful to the system receiving it as it was to the system from which it came.

continued on next page

CALS *(continued)*

CALS' philosophy is to make use of existing and emerging national and international standards for all aspects of the initiative. The CALS program regards the standards in the following diagram as essential to strategic planning efforts. The standards are divided into functional, interchange, data, and open standards (such as IRDS, GOSIP, SQL, RDA, and POSIX) necessary for information interchange in an open architecture and shared data base environment (see Figure 1). As technologies evolve and paradigms change, additional standards will be embraced by the CALS initiative.

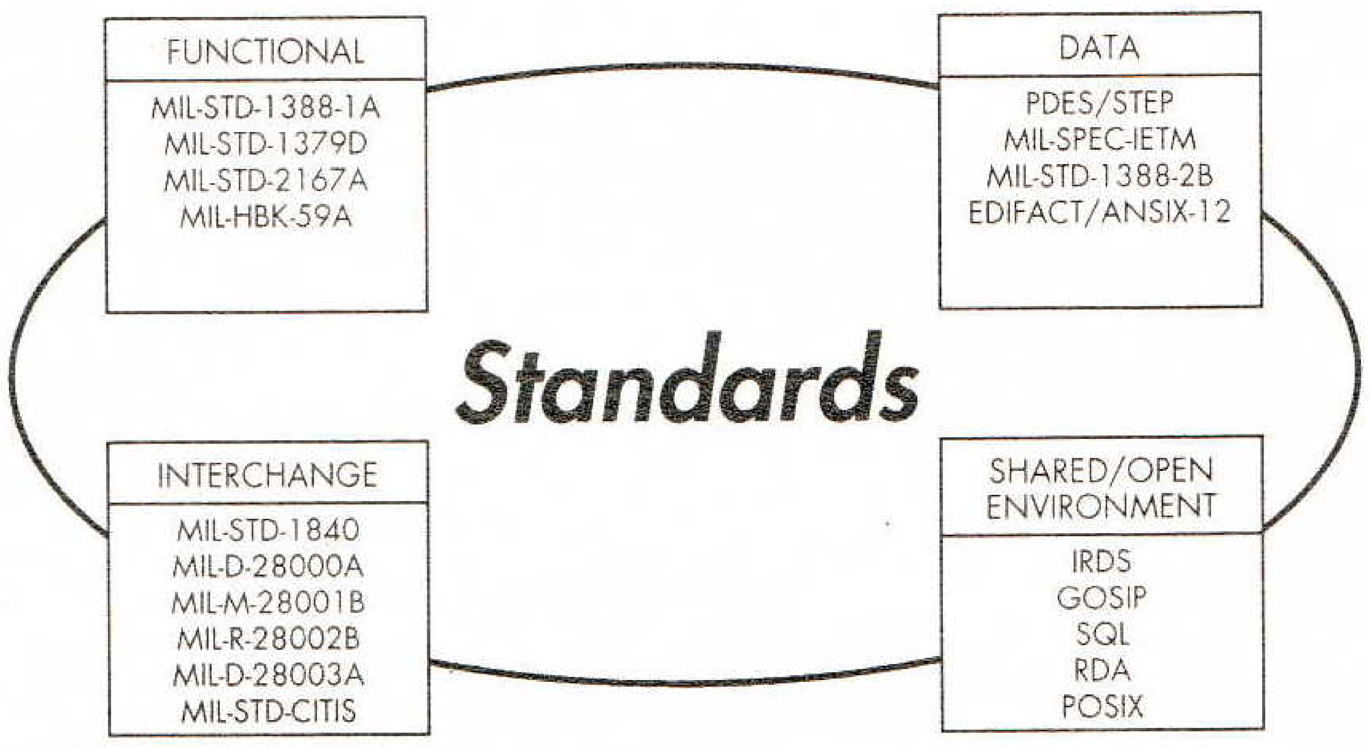


Figure 1: Standards critical to CALS

Functional standards

Functional standards include military standards and specifications and *Data Item Descriptions* (DIDs), which define functional processes, data requirements, data creation procedures, and the content and format of data products. Functional standards are used as a significant source of input when building applications to support user requirements.

Technical interchange standards

Technical standards define the medium and process of exchanging data between sending and receiving systems. These standards include federal standards, military standards and specifications, and other relevant conventions for the management, formatting, and physical media or telecommunications exchange of text, graphics, alphanumerics, and other forms of digital data. Figure 2 contains a brief description of the interchange standards.

Data standards

Data standards govern information sharing and data exchange in an open systems environment. These standards identify a specific set of data entities, and relationships among data entities and their attributes, which are often expressed in the form of a data dictionary and a set of rules that govern data definition, integrity and consistency. These standards also include file structure definitions, index keys, and other descriptive information needed for access to data bases.

The future of CALS

The future direction and vision of CALS involves continuous and timely access to information on products under development. The transition from simple digital CALS data interchange to total integration will necessitate a shared data environment. Instead of exchanging data between producers and consumers, data will be directly accessed by multiple users and organizations. Access will be available on geographically dispersed, heterogeneous hardware and software. Data access, delivery and distribution will be achieved through the *Contractor Integrated Technical Information Services* (CITIS) (MIL-STD-972). CITIS provides consumers with access to information associated with life-cycle product development. Product team partners will have authorized access to specific data to undertake their portion of the design, engineering or support of the product. Customers will also have access to these distributed databases for review of *Contractual Data Requirements List* (CDRL) items.

Standard	Basis	Description
MIL-STD-1840	Automated Interchange of Technical Information	defines organization and transfer of CALS data files
MIL-D-28000	ANSI Initial Graphics Exchange Specification (IGES) Standard Y14.26M	provides a mechanism for the definition of engineering data in the form of 2-D and 3-D vectors
MIL-M-28001	ISO Standard Generalized Markup Language (SGML) Standard number 8879	provides a mechanism for the definition of textual information, and external references to illustrations
MIL-R-28002	ISO Standard for scanned Raster images, number 8613, and the CCITT T-series Standard	provides a mechanism for the definition of monochrome bit-mapped images
MIL-D-28003	ISO Computer Graphics Metafile (CGM) Standard, ISO/IEC 8632:1992	provides a mechanism for the definition of technical illustrations in the form of 2-D vectors

Figure 2: Technical Interchange Standards

CALS and EDI

As mentioned earlier, technical data standards enable companies to format the same data formerly distributed on paper into electronic documents with defined structures. *Electronic Data Interchange* (EDI) is a maturing *business* philosophy which enables the exchange of business *and* technical data by employing a network protocol-based standard. This standard, referred to as ANSI ASC X12, utilizes X.400 and other network protocols as the mechanism to achieve data interchange. Other popular technologies, such as *value-added networks* (VANs) work with X12 to carry ASCII and binary CALS data to firms that conduct business electronically. The merging of CALS data with EDI transport mechanisms is revolutionizing purchases and other business transactions which involve technical data between buyers, contractors, and suppliers.

ANSI ASC X12 is currently used primarily throughout North America. The other dominant business data format standard, EDIFACT (*EDI for Administration, Commerce, and Transport*) is evolving as the international standard. In both of these standards, the components of a standardized electronic document are organized according to their specific functions. These documents, such as *Requests For Quote* (RFQ), invoice forms, purchase orders, etc., are known as *transaction sets* or messages. The X12 EDI transaction set Specification/Technical Information (841) facilitates exchange of CALS and other binary data.

A recent demonstration of the successful interrelationship of the CALS and X12 standards showed the transmission of digital engineering drawings over VANs using the X12 841 transaction set to small businesses. These small businesses downloaded the CALS data from their VAN mailboxes to their local PCs and viewed and manipulated the drawings locally for the purpose of preparing a response to an RFQ.

PDES/STEP

The international standard STEP (*Standard for the Exchange of Product Model Data*) is built on a concept of intelligent information structures designed to contain comprehensive product data, including geometric information, in a way that both the part definitions and part drawings can be generated.

continued on next page

CALS *(continued)*

STEP is being developed in an incremental manner which allows for the development and utilization of useful application-based subsets of the overall family of STEP parts. The ability to express in computerized or standard digital format all design, manufacture, and support information about a product will provide a complete, unambiguous, computer interpretable representation of a product throughout its life-cycle. CALS has a vital interest in the development and deployment of STEP. PDES (*Product Data Exchange using STEP*) is the U.S. Organizational activity that supports the development and implementation of STEP.

Concurrent Engineering

Concurrent Engineering (CE) is "...the systematic approach to integrated development of a product and its related processes, emphasizing responsiveness to customer expectations, and embodying team values of cooperation, trust, and sharing in such a manner that decision making proceeds with large intervals of parallel working by all life-cycle perspectives, synchronized by comparatively brief exchanges to produce consensus." [Concurrent Engineering Research Center (CERC), West Virginia University, Morgantown, WV, 1991.]

CALS is an enabler for CE by providing integrated development teams with correct, complete, accessible and timely digital product data. CALS and CE are inseparable for CE to be most productive.

Summary

CALS affects all aspects of manufacturing and business. Industry is leading the expansion of CALS, and the related technologies and concepts of EDI, STEP and Concurrent Engineering. With today's requirements of increased interoperability and data access, we should be taking a long, hard look at these concepts and considering how they can help us achieve our goals, while at the same time save money, speed data access, and improve quality.

References

[1] CALS Expo '92, *Proceedings of the 5th Annual Conference and Exposition,* December 7–10, 1992, San Diego: National Security Industrial Association, n.d.

[2] Lawson, Mike "Concurrent Engineering (CE) Overview," Tutorial delivered at the CE & CALS Washington '93 Conference and Exposition. Washington, June 28, 1993.

[3] U.S. Department of Defense, Office of the Secretary of Defense, *Computer-aided Acquisition & Logistic Support,* Washington: Office, December 1989.

[4] Stewart, J. "Components of OSI: Document Interchange Using ODA," *ConneXions,* Volume 5, No. 8, August 1991.

[5] Mark Bramhall and Jon A. Stewart, "Comparing Compound Document Processing Models," *ConneXions,* Volume 6, No. 8, August 1992.

There will be a BOF on this topic on Wednesday night at 7:30pm.

CAROLYN WIMPLE is Deputy Director of the CALS Test Network Office Test Bed at Lawrence Livermore National Laboratory. She has past experience working with the CGM and IGES standards at a detailed level. She played a key role in a recent test conducted by CTN/LLNL with the Sacramento Air Logistics Center at McClellan Air Force Base. This test demonstrated the transfer of CALS data, by using the ANSI X12 EDI standards, for the purpose of distributing RFQs to small businesses. She has also been active in analyzing MIL-STD-1840 and has proposed three amendments to 1840B. Carolyn was Chair of the CALS Solution Showcase Demonstration for INTEROP 93 Spring. The purpose of this demo was to educate the non-DoD public about the CALS strategy and its potential benefits to defense and non-defense related industry. Carolyn received her BS in Computer Science, with honors, in 1984 from California State University, Sacramento. She is a member of Upsilon Pi Epsilon, the Computer Science Honor Society. She can be reached as: `wimple@tis.llnl.gov`

Call for Papers

Introduction

The editors of *StandardView—The ACM Publication of Standardization,* invite contributions to this major new quarterly publication, to appear in September 1993. As the demand for standards becomes more widespread, the need for understanding becomes more critical. StandardView will focus on all aspects of standardization in the information technology industry—both theory and practice—with an emphasis on workaday utility. The journal will be eclectic and cross-disciplinary, attempting to represent the broadest range of the market available: both producers and users of standards in government, academia, and general interest groups.

Each edition of *StandardView* will have a primary focus, as well as a series of secondary articles covering general topics. Articles will range in length from five to 20 pages, and contributions on all aspects of standardization (including reasons not to standardize) are welcome. Because of the subject's nature, the editors are eager to acquire articles written in a style more similar to journalistic reporting than to scholarly research.

Topics

The topics for the initial editions include:

- *Human-Computer Interaction:* This issue will discuss theoretical and applied studies, standardization activities, applied ergonomics, and legislative and legal implications of ergonomics.
- *The Government and Standardization:* What is the government's role in standardization in Europe, North America, and the Pacific Rim? What are the social implications of standardization as an industrial policy tool?
- *Objects and Objectivity:* This issue will discuss the theory and practice of object based standardization, the economic rationale for objects and the need for standardization, and the implications of objects for information technology users and providers.
- *Open Systems as a Practical Discipline:* Papers will discuss the definition of Open Systems, and the efforts and discipline necessary to establish a true open system.
- *Internationalization and Localization:* Multilingual and multicultural software is increasingly important. Is standardization appropriate in this area?
- *Users and Standardization:* Do users care about standardization? Do current standards meet their needs? How can these needs be defined?

Submissions

Send papers or abstracts (500 words or less) to:

Carl Cargill, Editor-in-Chief
SunSoft
MTV 08-221
2550 Garcia Avenue
Mountain View, CA 94943
E-mail: carl.cargill@eng.Sun.com

More information

For subscription information, contact:

Member Services, ACM
1515 Broadway, 17th Floor
New York, NY 10036
Telephone: +1 212-626-0500
E-mail: ACMHELP@acm.org

CONNEXIONS
480 San Antonio Road
Suite 100
Mountain View, CA 94040
415-941-3399
FAX: 415-949-1779

FIRST CLASS MAIL
U.S. POSTAGE
PAID
SAN JOSE, CA
PERMIT NO. 1

ADDRESS CORRECTION
REQUESTED